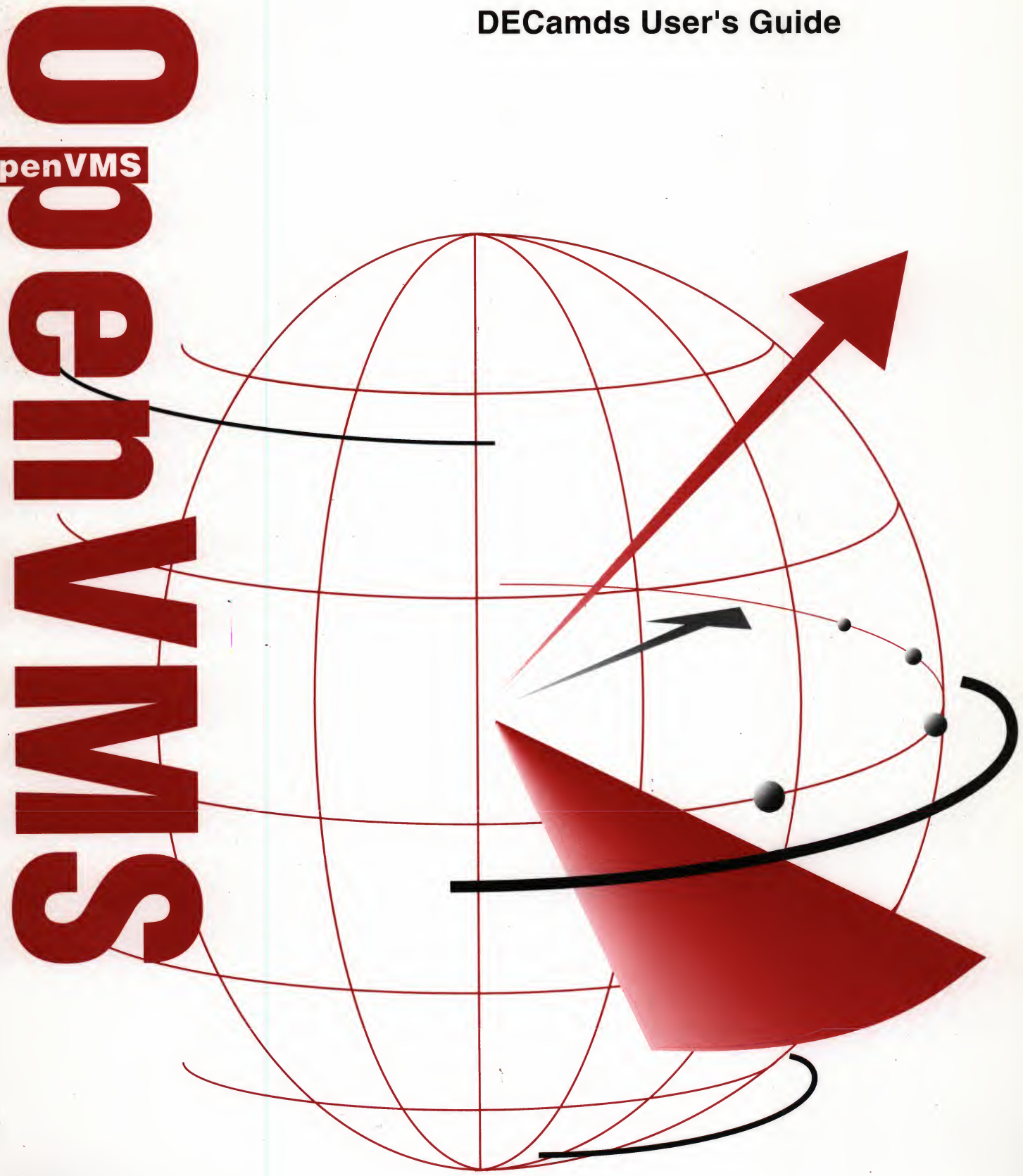
digital

DECamds User's Guide

OpenVMS



DECamds User's Guide

Order Number: AA–Q3JSB–TE

May 1995

This guide explains how to use DECamds software to detect and fix system availability problems. It also explains how to install DECamds.

Revision/Update Information:	This guide supersedes the *DECamds User's Guide*, Version 6.1.
Operating System and Version:	OpenVMS Alpha Version 6.2 OpenVMS VAX Version 6.2
Software Version:	DECamds Version 6.2

Digital Equipment Corporation
Maynard, Massachusetts

May 1995

ZK5929

This document is available on CD–ROM.

This document was prepared using VAX DOCUMENT Version 2.1.

Send Us Your Comments

We welcome your comments on this or any other OpenVMS manual. If you have suggestions for improving a particular section or find any errors, please indicate the title, order number, chapter, section, and page number (if available). We also welcome more general comments. Your input is valuable in improving future releases of our documentation.

You can send comments to us in the following ways:

- Internet electronic mail: `openvmsdoc@zko.mts.dec.com`
- Fax: `603-881-0120 Attn: OpenVMS Documentation, ZKO3-4/U08`
- Online form

  Print or edit the online form SYS$HELP:OPENVMSDOC_COMMENTS.TXT. Send the completed online form by electronic mail to our Internet address, or send the completed hardcopy form by fax or through the postal service.

  Please send letters or the form to:

  ```
  Digital Equipment Corporation
  Information Design and Consulting
  OpenVMS Documentation
  110 Spit Brook Road, ZKO3-4/U08
  Nashua, NH  03062-2698
  USA
  ```

Thank you.

How To Order Additional Documentation

Use the following table to order additional documentation or information. If you need help deciding which documentation best meets your needs, call 800-DIGITAL (800-344-4825).

Telephone and Direct Mail Orders

Location	Call	Fax	Write
U.S.A.	DECdirect 800.DIGITAL 800.344.4825	Fax: 800.234.2298	Digital Equipment Corporation P.O. Box CS2008 Nashua, NH 03061
Puerto Rico	809.781.0505	Fax: 809.749.8300	Digital Equipment Caribbean, Inc. 3 Digital Plaza, 1st Street, Suite 200 P.O. Box 11038 Metro Office Park San Juan, Puerto Rico 00910–2138
Canada	800.267.6215	Fax: 613.592.1946	Digital Equipment of Canada, Ltd. Box 13000 100 Herzberg Road Kanata, Ontario, Canada K2K 2A6 Attn: DECdirect Sales
International	—	—	Local Digital subsidiary or approved distributor
Internal Orders	DTN: 264.3030 603.884.3030	Fax: 603.884.3960	U.S. Software Supply Business Digital Equipment Corporation 10 Cotton Road Nashua, NH 03063–1260

ZK–7654A–GE

Contents

4 Performing Fixes

5 Customizing DECamds

A Installing DECamds

B Files and Logical Names

Examples

Figures

Tables

Preface

Intended Audience

This guide is intended for system managers who install and use DECamds software.

Document Structure

This guide contains the following chapters and appendixes:

- Chapter 1 describes an overview of DECamds software, including system software and hardware requirements, security features, and customizing security files.
- Chapter 2 describes how to start DECamds, using online help, the system overview window, and the event log window.
- Chapter 3 describes how to use the DECamds data windows.
- Chapter 4 describes how to take corrective actions, called **fixes**, to improve system availability.
- Chapter 5 describes the tasks you can perform to filter, sort, and customize the displaying of system data using DECamds. It also describes how some of these tasks can optimize the performance of DECamds.
- Appendix A contains instructions for installing DECamds.
- Appendix B contains a description of all files and logical names and gives examples of the log files that DECamds writes.
- The Glossary defines terms used for DECamds.

Related Documents

The following manuals provide additional information:

- *DECamds Version 6.2 Release Notes* describes features and changes that apply to DECamds software.
- *OpenVMS System Manager's Manual* describes tasks you perform to manage an OpenVMS system.
- *OpenVMS System Management Utilities Reference Manual* describes utilities you use to manage an OpenVMS system.
- *VMS DECwindows User's Guide* explains how to use DECwindows features.
- *Introduction to VMS System Services* explains OpenVMS lock management concepts.
- *OpenVMS License Management Utility Manual* explains how to use the License Management utility to register a product license.

- *OpenVMS System Messages: Companion Guide for Help Message Users* explains how to use Help Message.
- *POLYCENTER Software Installation Utility User's Guide* describes installing a product with the POLYCENTER Software Installation utility.

Conventions

The name of the OpenVMS AXP operating system has been changed to OpenVMS Alpha. Any reference to OpenVMS AXP or AXP are synonymous with OpenVMS Alpha or Alpha.

The following conventions are used to identify information specific to OpenVMS Alpha or to OpenVMS VAX:

Alpha	The Alpha icon denotes the beginning of information specific to OpenVMS Alpha.
VAX	The VAX icon denotes the beginning of information specific to OpenVMS VAX.
◆	The diamond symbol denotes the end of a section of information specific to OpenVMS Alpha or to OpenVMS VAX.

In this guide, every use of DECwindows and DECwindows Motif refers to DECwindows Motif for OpenVMS software.

The following conventions are used in this guide:

Ctrl/*x*	A sequence such as Ctrl/*x* indicates that you must hold down the key labeled Ctrl while you press another key or a pointing device button.
PF1 (GOLD) *x*	A sequence such as PF1 *x* indicates that you must first press and release the key labeled PF1 (defined as the GOLD key), then press and release another key or a pointing device button. The GOLD key definition is often mapped to the PF1 key on the keypad. GOLD key sequences can also have a slash (/), dash (–), or underscore (_) as a delimiter in EVE commands.
Return	In examples, a key name enclosed in a box indicates that you press a key on the keyboard. (In text, a key name is not enclosed in a box.)
. . .	A horizontal ellipsis in examples indicates one of the following possibilities: • Additional optional arguments in a statement have been omitted. • The preceding item or items can be repeated one or more times. • Additional parameters, values, or other information can be entered.
. . .	A vertical ellipsis indicates the omission of items from a code example or command format; the items are omitted because they are not important to the topic being discussed.

( )	In format descriptions, parentheses indicate that, if you choose more than one option, you must enclose the choices in parentheses.
[ ]	In format descriptions, brackets indicate optional elements. You can choose one, none, or all of the options. (Brackets are not optional, however, in the syntax of a directory name in an OpenVMS file specification, or in the syntax of a substring specification in an assignment statement.)
{ }	In format descriptions, braces indicate a required choice of options; you must choose one of the options listed.
boldface text	Boldface text represents the introduction of a new term or the name of an argument, an attribute, or a reason. Boldface text is also used to show user input in Bookreader versions of the manual.
italic text	Italic text indicates important information, complete titles of manuals, or variables. Italic text also represents information that can vary in system messages (for example, Internal error *number*), command lines (for example, /PRODUCER=*name*), and command parameters in text.
UPPERCASE TEXT	Uppercase text indicates a command, the name of a routine, the name of a file, or the abbreviation for a system privilege.
-	A hyphen in code examples indicates that additional arguments to the request are provided on the line that follows.
numbers	All numbers in text are assumed to be decimal, unless otherwise noted. Nondecimal radixes—binary, octal, or hexadecimal—are explicitly indicated.

1

Overview of DECamds

This chapter describes the following:

- Overview of DECamds
- System requirements for DECamds
- Security features

The Digital Availability Manager for Distributed Systems (DECamds) is a real-time monitoring, diagnostic, and correction tool that assists system managers to improve OpenVMS system and VMScluster availability. DECamds is also helpful to system programmers/analysts to target a specific node or process for detailed analysis, and system operators and service technicians to help determine hardware and software issues.

DECamds simultaneously collects and analyzes system data and process data from multiple nodes and displays the output on a DECwindows Motif display. Based on the analyzed data, DECamds detects events and proposes actions to correct resource availability and system denial issues in real time.

DECamds helps to improve OpenVMS system and VMScluster availability as follows:

Availability	Alerts users to resource availability problems, suggests paths for further investigation, and recommends actions to improve availability.
Centralized Management	Provides centralizing management of remote nodes within an extended local area network (LAN).
Intuitive Interface	Provides an easy to learn and use DECwindows Motif user interface.
Correction Capability	Allows real-time intervention, including adjustment of node and process parameters, even when remote nodes are hung.
Customization	Adjusts to site-specific requirements through a wide range of customization options.
Scalability	Makes job monitoring multiple OpenVMS systems and VMScluster systems easier over a single site or over multiple sites.

1.1 How Does DECamds Work?

DECamds is a client-server application. It is installed in two parts.

Data Provider	Gathers system data and transmits it to the Data Analyzer.
Data Analyzer	Receives data from the Data Provider, analyzes the data and displays it.

A node that has the DECamds **Data Provider** installed announces its availability, using a multicast LAN message, to any DECamds **Data Analyzer** that is installed and running. The Data Analyzer receives the Data Provider's availability announcement and a communications link is established.

Note

The Data Provider runs at a high interrupt priority level (IPL), so it will gather data and transmit it to the Data Analyzer, even if the Data Provider is on a remote node that is hung. However, the Data Provider cannot collect nonresident memory data, restricting some data collection in process space, because of the high IPL collection.

The Data Analyzer portion of DECamds is a DECwindows Motif application that runs on any OpenVMS system. Although you can run the Data Analyzer as a member of a monitored cluster, it is typically run on an OpenVMS system that is not a member of the cluster being monitored. You can have more than one Data Analyzer application executing in a LAN, but only one can be running on each OpenVMS system at the same time.

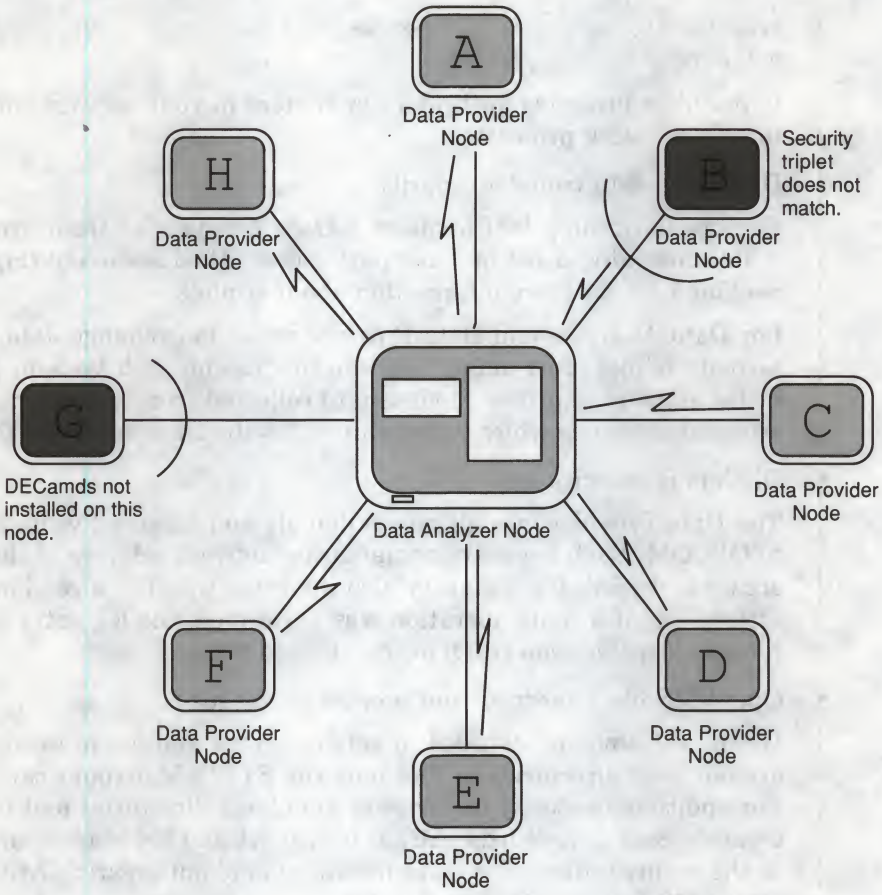
System data are analyzed and translated into meaningful values and rates that are displayed in DECwindows Motif windows. The data are screened for data points that exceed thresholds that may cause system or VMScluster availability problems. The Data Analyzer can also implement various system correction options if authorized to do so.

The Data Analyzer and Data Provider nodes communicate over an Extended LAN using an IEEE 802.3 Extended Packet format protocol. Once a secure connection is established, the Data Analyzer instructs the Data Provider to gather specific system and process data.

Figure 1–1 illustrates the interaction of the Data Analyzer and Data Provider on nodes in a cluster.

Nodes A, C, D, E, F, and H can exchange information with the Data Analyzer. Node B has defined its security to exclude the Data Analyzer from accessing its system data. Node G has not installed DECamds and does not communicate with the Data Analyzer.

Figure 1–1 DECamds Processing



1.2 System Requirements for DECamds

This section is an overview of DECamds system requirements. Appendix A describes the specific system hardware and software requirements for installing and running the DECamds Data Analyzer and Data Provider.

Any system that is supported by OpenVMS can install the DECamds Data Provider.

The DECamds Data Analyzer can be run from either a cluster member or a standalone system outside the cluster. However, Digital recommends that you run the Data Analyzer from outside a cluster because then you can monitor system information even if the nodes in the cluster pause or hang.

System requirements vary with the number of nodes to be monitored and the amount of data to be collected. See Appendix A for more information.

1.3 Security Features

DECamds has several security features including the following:

- Private LAN transport

The DECamds protocol is based on the 802.3 Extended Packet Format (also known as SNAP). The IEEE DECamds protocol values are as follows:

```
Protocol ID:        08-00-2B-80-48
Multicast Address:  09-00-2B-02-01-09
```

If you filter protocols for bridges or routers in your network, add these values to your network protocols.

- DECamds data transfer security

  Each node running DECamds as a Data Analyzer or Data Provider has a file containing a list of three-part codes, called **security triplets**. See Section 1.3.1 for more information about triplets.

  For Data Analyzer and Data Provider nodes to exchange data, at least one security triplet must match between the files on each system. Most DECamds nodes allow read access to view data collected from a node. Setting write access determines which nodes can act as the Data Analyzer to perform fixes.

- DECamds security log

  The Data Provider logs all access denials and executed write instructions to OPCOM. Each log entry contains the network address of the initiator. If access is denied, the log entry also indicates whether a read or write was attempted. If a write operation was performed, the log entry indicates the process identification (PID) of the affected process.

- OpenVMS file protection and process privileges

  When DECamds is installed, it sets directory and file protections on its system level directories so that only the SYSTEM account can read the files. For additional security on these system level directories and files, you can create access control lists (ACLs) to restrict and set alarms on write access to the security files. For more information about creating ACLs, see the *OpenVMS Guide to System Security*.

The AMDS$CONFIG logical translates to the location of the default security files, including the following:

- The AMDS$DRIVER_ACCESS.DAT file is installed on all Data Provider nodes. The file contains a list of Data Analyzer nodes to which system data can be sent. It also contains the type of access allowed for each of those nodes.
- The AMDS$CONSOLE_ACCESS.DAT file is installed on only those nodes that will run the Data Analyzer portion of DECamds. It contains a list of passwords to identify itself to Data Provider nodes.

You can create additional security files in the directory associated with the AMDS$CONFIG logical name. By default, this logical name is assigned to AMDS$SYSTEM. As you customize DECamds, you can change the logical assignment of AMDS$CONFIG to read input files from other locations.

The following sections describe what a security triplet is, where to find the security files, and how to set up your security files.

1.3.1 Understanding DECamds Security Files

A security triplet determines which systems can access system data from the node. Security triplets are listed in the AMDS$DRIVER_ACCESS.DAT and AMDS$CONSOLE_ACCESS.DAT files on the Data Analyzer and Data Provider systems.

A security triplet is a three-part record which is separated by backslashes (\ ) in either the AMDS$DRIVER_ACCESS.DAT file or the AMDS$CONSOLE_ACCESS.DAT file. A triplet consists of the following fields:

- A network address (DECnet address, hardware address, or a wildcard character)
- An 8-character (alphanumeric) password

  The password is not case-sensitive, so the passwords "testtest" and "TESTTEST" are considered to be the same.
- A read or write (R or W) access verification code

  The only restriction for the Data Analyzer is that the security triplets that allow write access are listed last in the AMDS$CONSOLE_ACCESS.DAT security file.

The exclamation point (!) is a comment delimiter; any characters after the comment delimiter are ignored.

Table 1–1 describes the detailed format of each portion of the security triplet and then gives some examples for different situations.

Table 1–1 Security Triplet Format

Item	Description
DECnet Address (area.number)	Although DECnet is not required to run DECamds, the DECnet address is used to determine a node's physical address. The DECnet address is created by using the *area.number* format, where area is a value from 1 to 63, and number is a value from 1 to 1023. This address is modified into a physical address of the form AA-00-04-00-xx-yy to conform to the standard IEEE 802.3 protocol for network addressing. The AA-00-04-00 prefix is associated with the Digital owned address. The xx-yy suffix is the hexadecimal representation of the address formula: `area*1024+number` **Note** If you are running on a system with more than one LAN adapter or are running DECnet/OSI networking software, then this format is not valid for you. You must then use the Hardware Address or Wildcard Address format for this field.

(continued on next page)

Table 1–1 (Cont.) Security Triplet Format

Item	Description

Hardware Address (08-00-2B-xx-xx-xx)	The hardware address field is the physical hardware address in the LAN adapter chip. It is used if you have multiple LAN adapters or are running the DECnet/OSI networking software on the system (as opposed to the DECnet Phase IV networking software).
	For adapters provided by Digital, the hardware address is in the form 08-00-2B-xx-xx-xx, where the 08-00-2B portion is Digital's valid range of LAN addresses as defined by the IEEE 802 standards and the xx-xx-xx portion is chip specific.
	To determine the value of the hardware address on a system, use the OpenVMS System Dump Analyzer (SDA) as follows:
	`$ ANALYZE/SYSTEM`
	`SDA> SHOW LAN`
	The previous commands display a list of available devices. Choose the template device of the LAN adapter you will be using and then enter the following command:
	`SDA> SHOW LAN/DEVICE=xxA0`

Wildcard Address (*)	The wildcard character allows any incoming triplet with a matching password field to access the Data Provider node. Use the wildcard character to allow read access and to run the console application from any node in your network. Because the Data Analyzer does not use this field, you should use the wildcard character in this field in the AMDS$CONSOLE_ACCESS.DAT file.

Caution

Use extreme caution if you intend to use the wildcard character for write security triplets because this enables any system to perform system-altering fixes.

The following steps show how the security triplets are used by DECamds to ensure security among DECamds nodes:

1. A message is broadcast at regular intervals to all nodes within the LAN to indicate the availability of a node to communicate with a node that is running DECamds to gather system data.
2. The node running the Data Analyzer receives the availability message and returns a security triplet, identifying itself to the Data Provider, and requests system data from the Data Provider.
3. The security triplet is examined by the Data Provider to determine if the Data Analyzer is listed in the AMDS$DRIVER_ACCESS.DAT file to permit access to the system.
   - If the Data Analyzer access information is listed in the AMDS$DRIVER_ACCESS.DAT file, then the Data Provider and the Data Analyzer can exchange information.

- If the Data Analyzer is not listed in the AMDS$DRIVER_ACCESS.DAT file, or does not have appropriate access information, then access is denied and a message is logged to OPCOM, and the Data Analyzer receives a message stating that access is not permitted to that node.

Table 1–2 describes how the Data Provider node interprets a security triplet match.

Table 1–2 Security Triplet Verification

Security Triplet	Interpretation
08-00-2B-12-34-56\HOMETOWN\W	The Data Analyzer has write access to the node only when the Data Analyzer is run from the node with this hardware address (multi-adapter or DECnet/OSI system) and with the password HOMETOWN.
2.1\HOMETOWN\R	The Data Analyzer has read access to the node when run from a node with DECnet Phase IV address 2.1 and the password HOMETOWN.
*\HOMETOWN\R	Any Data Analyzer with the password HOMETOWN has read access to the node.

1.3.2 Customizing Security Files

Security files define which Data Analyzers can access data on nodes that have a Data Provider. The security files let you group nodes according to specific criteria.

Digital recommends you group nodes according to VMScluster membership. A node can be in only one group at a time. Installing DECamds initially assigns all nodes to one group. Each node that is assigned to a group is listed under the group name heading in the System Overview window.

Consider the following items when you set up customized groups:

- VMSclusters and data integrity
  - For data in the disk volume and lock contention windows to be complete and accurate, all nodes in a VMScluster must be in the same group.

    It is possible to include two clusters in one group, but if a cluster is divided between two groups or only partially included, the data may not be accurate.
  - Adding standalone nodes to the group does not affect the accuracy of the data, as long as the entire VMScluster is in the same group. For sites where disk volume and lock contention data are not critical, this might be acceptable.
- Partitioning for analysis

  Specific users can have read or write access to certain subsets of nodes. For example, one Data Analyzer can be designated to monitor a certain hardware type or cluster. *This is entirely independent of the group to which the nodes of that hardware type or cluster are assigned.* Apart from strict security considerations, this mechanism is often used to partition systems for convenience.

  Your site might already have criteria relevant to defining groups. These could include a system management division of labor, hardware type, physical location, or work function.

Digital recommends that you correlate your security files to your group definitions so that all nodes in the group are visible in the System Overview window. Section 1.3 explains how to set up security files.

1.3.2.1 Setting Up Node Groups

To assign a node to a group, perform the following steps on each Data Provider node that is to be part of the group:

1. Assign a unique name of up to 15 alphanumeric characters to the AMDS$GROUP_NAME logical name in the AMDS$SYSTEM:AMDS$LOGICALS.COM file. For example:

```
$ AMDS$DEF AMDS$GROUP_NAME FINANCE ! Group FINANCE; VMScluster alias
```

2. Apply the logical name by restarting the Data Provider, as follows:

```
$ @SYS$STARTUP:AMDS$STARTUP.COM START
```

For more information about the other logical names contained in AMDS$LOGICALS.COM, see Appendix B.

1.3.2.2 Defining Data Exchange Access Between Nodes

The Data Provider stores access security triplets in a file called AMDS$DRIVER_ACCESS.DAT, which indicates the Data Analyzer nodes that are allowed request their data to be provided; if a Data Analyzer node is not listed in the file, access is denied.

For example, all Data Provider nodes in Group FINANCE have the following AMDS$DRIVER_ACCESS.DAT file:

AMDS$DRIVER_ACCESS.DAT file for Group FINANCE:

```
*\FINGROUP\R   ! Let anyone with FINGROUP password read
               !
2.1\DEVGROUP\W ! Let only DECnet node 2.1 with
               ! DEVGROUP password perform fixes (writes)
               !
2.2\FINGROUP\W ! Let DECnet node 2.2 perform fixes
```

For example, all Data Provider nodes in Group DEVELOPMENT have the following AMDS$DRIVER_ACCESS.DAT file:

AMDS$DRIVER_ACCESS.DAT file for Group DEVELOPMENT:

```
*\GROUPBRD\R   ! Let anyone with GROUPBRD password read
               !
2.1\DEVGROUP\W ! Let only DECnet node 2.1 with
               ! DEVGROUP password perform fixes
```

AMDS$CONSOLE_ACCESS.DAT file for a Data Analyzer

For a Data Analyzer to access information on any node in Groups FINANCE or DEVELOPMENT, the following access security triplets must be listed in the Data Analyzer node's AMDS$CONSOLE_ACCESS.DAT file:

```
*\FINGROUP\R ! To access data on nodes in Group FINANCE
             !
*\GROUPBRD\R ! To access data on nodes in Group DEVELOPMENT
             !
*\DEVGROUP\W ! Assumes you are the owner of DECnet
             ! address 2.1 so you can access data and
             ! perform fixes on both Group FINANCE and
             ! Group DEVELOPMENT nodes.
             !
*\FINGROUP\W ! Assumes you are the owner of DECnet
             ! address 2.2 so you can access data and
             ! perform fixes on Group FINANCE nodes.
```

After you modify the AMDS$CONSOLE_ACCESS.DAT security file, restart the Data Analyzer with the AVAIL command to use the changes. For more information about starting DECamds, see Chapter 2.

1.3.2.3 Defining Read Access for Specific Users

You can restrict the write access to certain users by performing the following steps:

1. Assign a search list of directories to the AMDS$CONFIG logical name in the AMDS$SYSTEM:AMDS$LOGICALS.COM file. For example:

   ```
   $ DEFINE AMDS$CONFIG SYS$LOGIN,AMDS$SYSTEM
   ```

   Execute the procedure as follows:

   ```
   $ @AMDS$SYSTEM:AMDS$LOGICALS
   ```

2. Copy the AMDS$CONSOLE_ACCESS.DAT security file to the SYS$LOGIN directory of a user and edit the file for that user.
3. Restart the Data Analyzer with the AVAIL command. For more information about starting the Data Analyzer, see Chapter 2.

The next time the user starts DECamds, the new security file will be found in their SYS$LOGIN directory and will be used. The security file found in AMDS$SYSTEM will not be read.

1.3.3 Sending Messages to OPCOM

The logical names shown in Table 1–3 control the sending of messages to OPCOM and are defined in the AMDS$LOGICALS.COM file.

Table 1–3 DECamds Logical Names for OPCOM Messages

AMDS$RM_OPCOM_READ	A value of TRUE logs read failures to OPCOM.
AMDS$RM_OPCOM_WRITE	A value of TRUE logs write failures to OPCOM.

To use the changes, restart the Data Analyzer with the following command on each system or use SYSMAN to run the command on all systems within the VMScluster:

```
$ @SYS$STARTUP:AMDS$STARTUP RESTART
```

1.3.4 Setting Broadcast Intervals for Node Availability Messages

Availability messages are broadcast by the Data Provider on nodes at regular intervals until a node establishes a link with the Data Analyzer. After a link has been established, the interval varies depending on the amount of data collection (and other factors) occurring between nodes.

You can modify the logical names in the AMDS$LOGICALS.COM file (shown in Table 1–4) to change the broadcast availability intervals.

Table 1–4 Broadcast Availability Logical Names

AMDS$RM_DEFAULT_INTERVAL	Defines from 15- to 300-second intervals between availability message broadcasts.
AMDS$RM_SECONDARY_INTERVAL	Defines from 15- to 1800-second intervals between availability message broadcasts after a link has been established between nodes.

To use the changes, restart the Data Analyzer with the following command on each system or by using SYSMAN to run the command on all systems within the VMScluster:

```
$ @SYS$STARTUP:AMDS$STARTUP RESTART
```

2

Getting Started

This chapter describes the following:

- How to start DECamds
- How to use the System Overview windows to monitor resource availability problems on your system
- How to use the Event Log window to correct resource availability problems on your system

2.1 Starting DECamds

To start the DECamds Data Analyzer, enter the following command and any of the following qualifiers that you want:

AVAIL /qualifiers

Qualifiers

/CONFIGURE
Specifies the directories from which input files are read. This may be a search list of directories or a logical defining a search list of directories.

/LOG_DIRECTORY
Specifies the directory to which log files are written. Output files can be directed to the null device, NLA0:.

/GROUP
A comma separated list of the groups of Data Provider nodes you want the Data Analyzer to access.

Note

If you have not already set up a group hierarchy of nodes during DECamds installation, refer to Section 1.3.2 for information about setting up node groups.

The following commands start DECamds with input files read first from SYS$LOGIN, then from AMDS$SYSTEM (if not found in SYS$LOGIN). All output files are written to the SYS$LOGIN directory. Only data from group KUDOS is collected.

```
$ DEFINE/JOB AMDS$CONFIG  SYS$LOGIN,AMDS$SYSTEM
$ AVAIL/CONFIGURE=AMDS$CONFIG/LOG_DIRECTORY=SYS$LOGIN/GROUP=(KUDOS)
```

When DECamds starts, it displays the System Overview and Event Log windows.

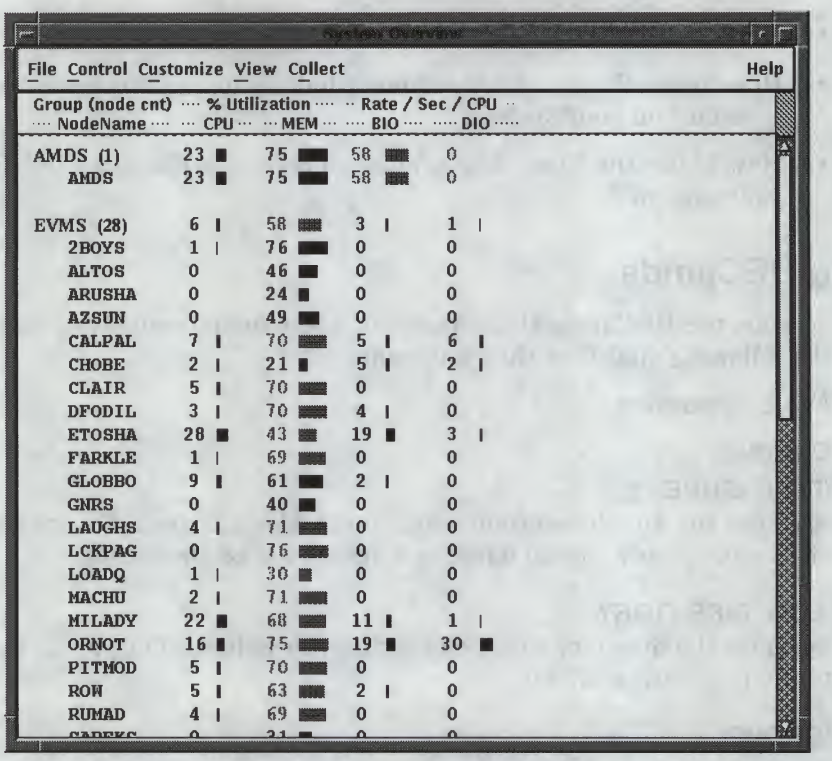
To obtain help about DECamds, choose a menu item from the Help menu.

A glossary of terms is listed in the Glossary.

2.2 Using the System Overview Window

The System Overview window allows you to focus on resource usage activity at a high level and display more specific data when necessary. The System Overview window displays CPU, memory, and I/O data for each node and group it recognizes. Figure 2–1 shows a sample System Overview window.

Figure 2–1 System Overview Window



The System Overview window contains two kinds of information:

- Group information, displayed in the row next to the Group name, shows averages for all nodes in the group.
- Node information, displayed in the row next to the node name, shows averages for the node.

  If the View menu is set to Hide Nodes, node information is not displayed.

Table 2–1 explains the fields displayed in the System Overview window.

Table 2–1 System Overview Window Display Fields

Field	Description
Group	Displays the group names in alphabetical order and the number of nodes recognized by DECamds. A group is a defined set of nodes that appear together in the System Overview window. A group can be defined by type of hardware, physical location, function, or VMScluster alias.
NodeName	Displays the name of the node in a node row.
CPU (CPU usage)	In a group row, displays the average of the percentage of CPU time used by all processors weighted toward the present. In a node row, displays the percentage of CPU time used by all processes on the node, expressed as an exponential average, weighted toward the present. On Symmetric Multiprocessing (SMP) nodes, rates for CPU time are added and divided by the number of CPUs.
MEM (Memory rate)	In a group row, displays the average of the sampled values (over time) for all processes on all nodes in a group. In a node row, displays the percent of space in physical memory that all processes on the node are currently occupying. The value represents 100 percent minus the amount of free memory.
BIO (Buffered I/O rate)	In a group row, displays the average of BIO operations of all processes on all nodes. In a node row, displays the BIO rate for all processes on the node across the number of CPUs.
DIO (Direct I/O usage)	In a group row, displays the average of DIO operations of all processes on all nodes. In a node row, displays the DIO rate for all processes on the node.

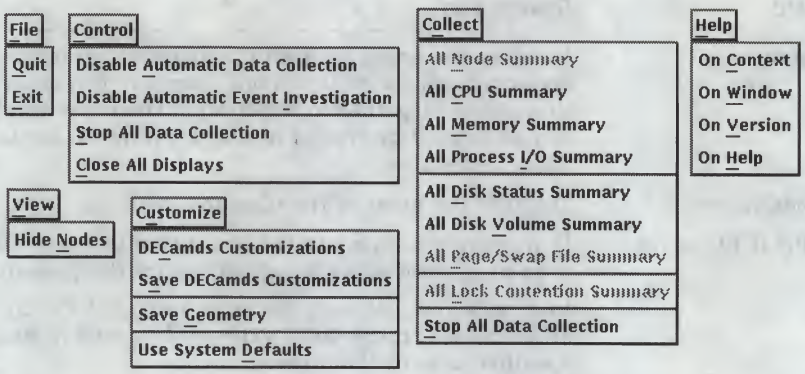
A percentage of a used resource is shown both by number and a dynamic status bar. For group rows, the values are averaged for all nodes in the group when Node Summary collecting is active.

Resource availability problems are indicated by highlighting. When an event occurs, DECamds highlights the status bar that represents the resource. Highlighting is shown in red on color monitors, by default; bold on monochrome monitors. You can change the highlight color. (See Chapter 5 for more information.)

When data appears dimmed, the data is more than 60 seconds old due to a user action that stopped Node Summary data collection. When the data is updated, the display returns to normal resolution.

Figure 2–2 shows the System Overview window options.

Figure 2–2 System Overview Window Menus



2.2.1 Expanding and Collapsing Group Information

Use the View menu to display **group** or group and node status in the System Overview window. Typically a group is a VMScluster. Groups are displayed in alphabetical order. Nodes within a group are also displayed in alphabetical order.

You can also expand and collapse specific group displays by pressing MB3 while the cursor is on the selected group and choosing either the Hide Nodes or Show Nodes menu item.

2.2.2 Displaying Additional Data

By default, the Data Analyzer collects, analyzes, and displays three categories of data from the Data Provider nodes:

- Node Summary
- Lock Contention Summary
- Page/Swap File Summary

In addition to the default data, you can choose any of these categories of additional data to be collected, analyzed, and displayed:

- CPU Summary
- Memory Summary
- Process I/O Summary
- Disk Status Summary
- Disk Volume Summary

You can change the default data windows with the DECamds Application Customization dialog box. For more information about customizing DECamds, see Chapter 5.

Note

Data gathering and display consume CPU time and network bandwidth. Request only the data you need to conclude an investigation, then stop collecting the data (see Section 2.2.3). Whenever possible, just collect data for a node, not the entire group.

To request a specific data category, perform one of the following steps:

- From the Event Log window, press MB3 over a selected event and choose Display from the menu.
- For data on a single node or a group, from the System Overview window, press MB3 over a selected node or group, choose Collect from the menu, and choose a category from the submenu.
- For data on all nodes, from the System Overview window, choose a category from the Collect menu.

2.2.3 Stopping Data Collection

To stop collecting data, choose one of the following options:

- Stop All Data Collection from the Collect menu or Control menus of the System Overview or Event Log windows.

  This stops collecting for all nodes. Events are removed from the Event Log, and data values in the System Overview window go to zero and are dimmed. Use this item if you lose track of data you are collecting in the background. Then restart data collection as needed; new events appear once data collection resumes.

- Stop All Data Collection from the Collect submenu of the MB3 menu for a group or node name in the System Overview window.

  This stops all data collection for the group or node you select. Node or group data in the System Overview window is zeroed.

- Stop Collecting from the File menu of any data window.

  If the data window is specific to a node or group, this option stops collecting for the node or group.

  Note

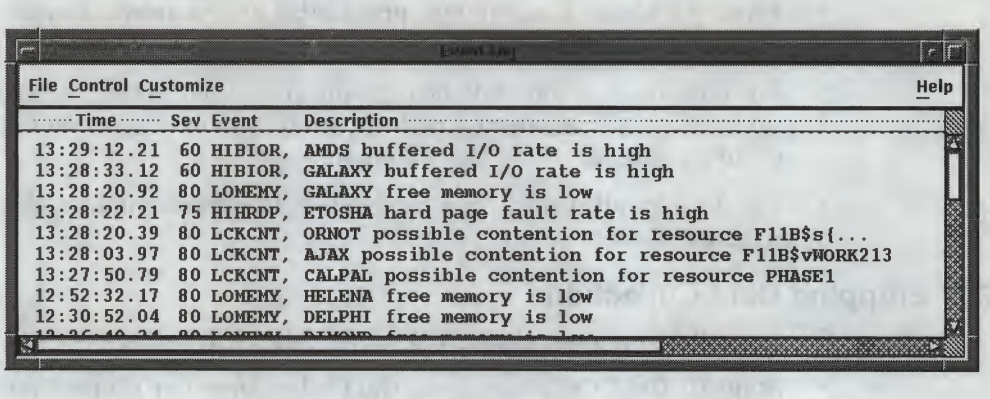
  Close Display in the File menu closes the window but continues data collection as a background task.

- Exit or Quit from the File menu.

2.3 Using the Event Log Window

The Event Log window (Figure 2–3) allows you to identify and correct a system problem. The Event Log window displays a warning message whenever DECamds detects a resource availability problem.

Figure 2–3 Event Log Window



Note

DECamds writes all events to a log file (AMDS$LOG:AMDS$EVENT_LOG.LOG). You can read this file while the application is running.

Ignore event messages that report the system process "SWAPPER" as having used all its quotas. The SWAPPER process is the OpenVMS memory management process; it does not have its quotas defined in the same way other system and user processes do.

Table 2–2 explains the fields displayed in the Event Log window.

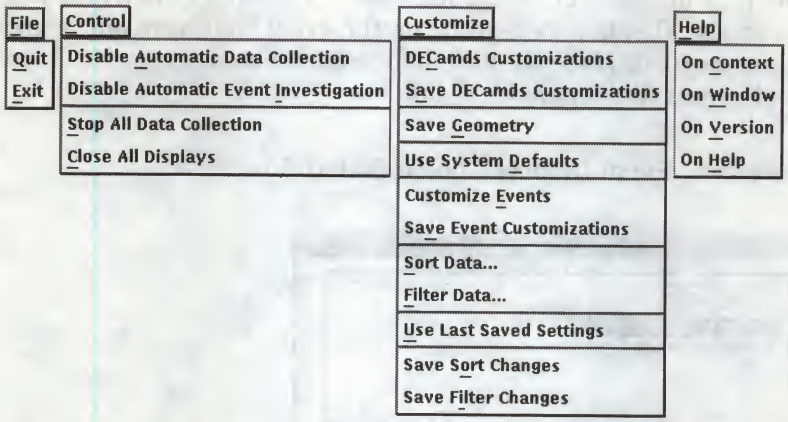
Table 2–2 Event Log Window Display Fields

Field	Description
Time	Displays an event, in real time, when detected.
Severity	Displays a value from 0 to 100. By default, events are listed in the Event Log window in order of decreasing severity. 0 is an informational message; 100 is a severe event. An event severity of 80 is high and indicates a potentially serious problem. Events with a severity of less than 50 appear dimmed, to indicate that they are less important. See Chapter 5 for information about how to change the severity. Events that are critical are also sent to the OpenVMS operator communication manager (OPCOM).
Event	Displays an alphanumeric identifier of the type of event.
Description	Displays the node or group name and a short description of the resource availability problem.

When an event "times out" by an improvement in availability, it is removed from the display. Events that are not triggered by a condition are timed out after 30 seconds (for example, the "CFGDON, *node* configuration done" event). When you select an event, the event remains displayed for 15 seconds (or until you initiate another task in the window), even if the event times out.

Figure 2–4 shows the Event Log window options.

Figure 2–4 Event Log Window Menus



To customize the Event Log window display, choose Filter Data... from the Customize menu. Table 2–3 describes the filter options.

Table 2–3 Event Log Filters

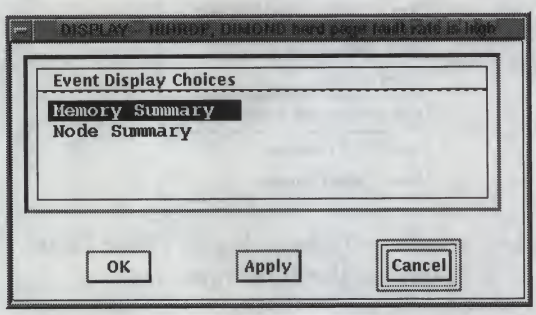
Filter	Description
Event Bell	Determines which events are marked by an audible signal by specifying a minimum event severity value. When a new event is displayed, if the severity value is the same or greater than the specified value, an audible notification is given. To disable the sound, specify a value of 101.
Event Highlight	Determines which events are marked by a visual signal by specifying a minimum event severity value. When a new event is displayed, if the severity value is the same or greater than the specified value, an event is highlighted. To disable highlighting, specify a value of 101.
Event Signal	Determines which events get displayed in the Event Log window by specifying a minimum event severity value. When a new event is received, if the severity value is the same or greater than the specified value, it is displayed; events with a severity level less than the specified value are ignored.
Event Timeout (secs)	Determines how long an informational event is displayed (in seconds).
Event Escalation Time (secs)	Determines how long an event must be signaled before it is sent to the operator communication manager (OPCOM). DECamds uses this value along with the Event Escalation Severity value. Both criteria must be met before the event is signaled to OPCOM.
Event Escalation Severity	Determines which events are sent to the operator communication manager (OPCOM). DECamds uses this value along with the Event Escalation Time (secs) value. Both criteria must be met before the event is signaled to OPCOM.

For more information about customizing event log information, see Section 5.1.

2.3.1 Displaying Information About an Event Log Entry

To display information, press MB3 on any event in the Event Log window, and then choose Display. Depending on the event, you have one or more event display choices that will give you more information about the event. Figure 2–5 shows a sample event display choice dialog box.

Figure 2–5 Event Display Choice Dialog Box



2.3.2 Performing Corrective Action on an Event Log Entry

To take corrective action, press MB3 on any event in the Event Log window, and then choose Fix. Depending on the event, you have one or more of the following event fix choices. (Not all events have all fix options.)

Adjust process working set
Crash node
Delete a process
Exit an image
Lower process priority
Purge process working set
Raise process priority
Resume a process
Suspend a process

See Chapter 4 for detailed information about performing fixes.

2.3.3 Sending Event Information to OPCOM

DECamds sends critical events to the Operator Communication Manager (OPCOM) so that POLYCENTER products may be able to access the events.

By default, events that meet *both* of the following criteria are sent to OPCOM:

- A severity level of 90 or above
- Occur continuously for 600 seconds (10 minutes)

You can change either criterion by choosing Filter Data... from the Customize menu of the Event Log window. For more information on changing Event Log filters, see Chapter 5.

2.3.4 Removing an Event from the Event Log Window

To remove an event from the Event Log window, press MB3 on an event, and choose Remove from the menu. An event will reappear if DECamds routine sampling detects the same situation that caused the original log entry.

2.3.5 Retaining and Releasing an Event in the Event Log Window

Event Log entries are removed when the underlying cause is removed, so an event may disappear from the Event Log window. To retain the selected event in the Event Log window, press MB3 on an event and choose Freeze. When an event is frozen, the Time field is highlighted.

To release the selected event, press MB3 on the event and choose Unfreeze.

3

Managing DECamds Data Windows

This chapter describes the DECamds data windows that you can display from the System Overview and Event Log windows. Table 3–1 shows the data windows and their function.

Table 3–1 DECamds Data Windows

Window	Opened from Window	Displays
CPU Modes Summary Section 3.1	Event Log Node Summary System Overview	A graphic representation of each CPU's processor modes, listing the process currently executing in the CPU.
CPU Summary Section 3.2	Event Log Node Summary System Overview	Statistics about CPU utilization by process, including process state, priority, execution rate, CPU time, and wait time.
Disk Status Summary Section 3.3	Event Log System Overview	Disk device data including path, volume name, status, and mount, transaction, error, and resource wait counts.
Disk Volume Summary Section 3.4	Event Log System Overview	Disk volume data, including path, volume name, disk block utilization, queue length, and operation count rate.
Lock Contention Summary Section 3.5	Event Log System Overview	Data about each resource for which a potential lock contention situation exists.
Memory Summary Section 3.6	Event Log Node Summary System Overview	Statistics about memory usage by process, including process working set count, quota and extent, and paging rates.
Node Summary Section 3.7	Event Log System Overview	Overview of a specific node's resource demand on the CPU state queues and processor modes, memory utilization, page faults, and I/O.
Page/Swap File Summary Section 3.8	Event Log System Overview	Data about page and swap names and utilization, including free, used, and reserved pages.
Process I/O Summary Section 3.9	Event Log Node Summary System Overview	Statistics about I/O utilization by process, including Buffered I/O, Direct I/O, Page Write I/O, and also lists various I/O quotas.
Single Lock Summary Section 3.10	Lock Contention Summary Window	Specific data about the blocking lock and any other locks in the granted, conversion, or waiting queues.

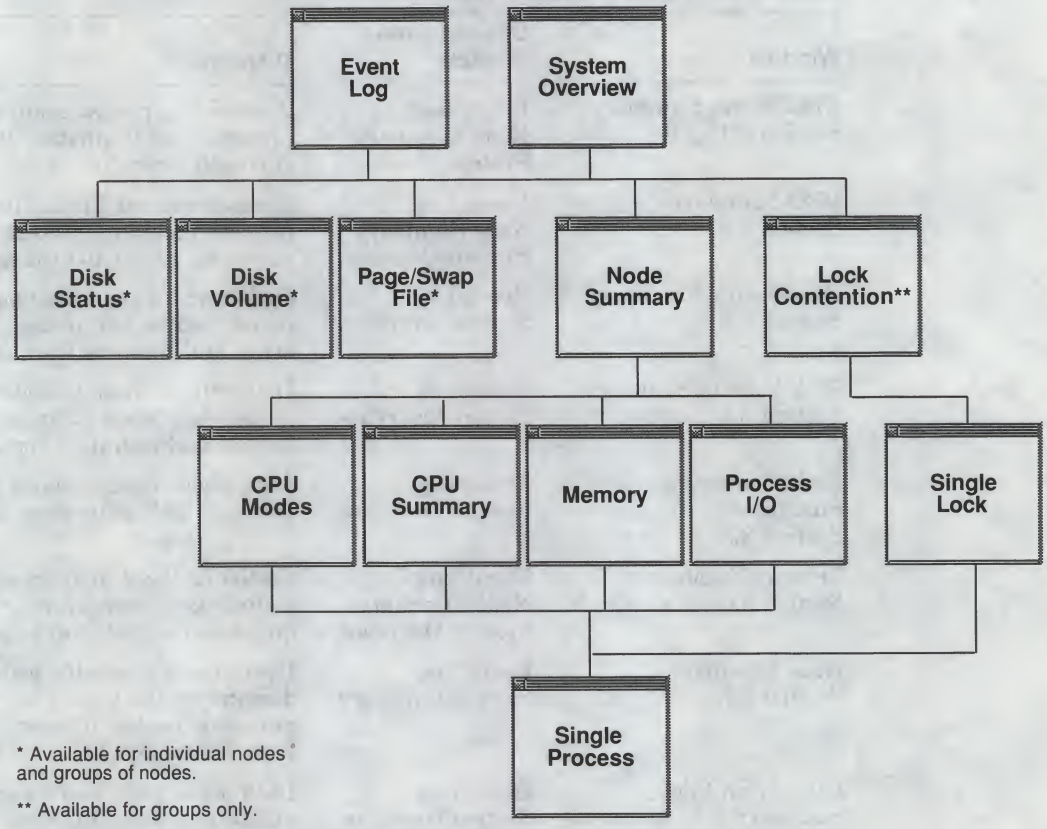
(continued on next page)

Table 3–1 (Cont.) DECamds Data Windows

Window	Opened from Window	Displays
Single Process Summary Section 3.11	Any Data Window	Specific data about a process, basically a combination of data elements from the CPU, Memory, and Process I/O displays, as well as data for specific quota utilization, current image, specific process information, and wait queue time.

Figure 3–1 shows the hierarchy of the DECamds data windows.

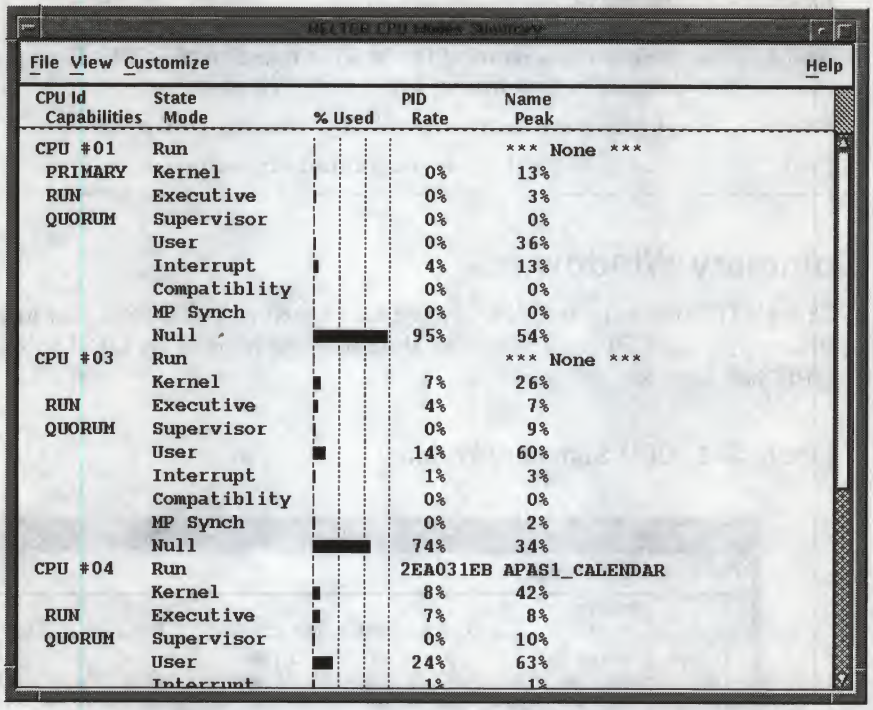
Figure 3–1 DECamds Data Window Hierarchy



3.1 CPU Modes Summary Window

The CPU Modes Summary window shown in Figure 3–2 displays more detailed summary statistics about CPU mode usage than the Node Summary window. Use the CPU Modes Summary window to diagnose issues that may be caused by CPU-intensive users or CPU bottlenecks.

Figure 3–2 CPU Modes Summary Window



To open a CPU Modes Summary window, perform one of the following steps:

- From the Node Summary window, double click on the CPU Modes area. You can also press MB3 and choose Display Modes from the menu.
- From the View menu, choose Display CPU Modes Summary.

You can open a window about a specific process in the CPU Modes Summary window by double clicking on the process name.

Table 3–2 describes the CPU Modes Summary window data fields.

Table 3–2 CPU Modes Summary Window Data Fields

Field	Displays
CPU ID	A decimal value representing the identity of a process in a multiprocessing system. On a uniprocessor, this value will always be CPU #00.
Capabilities	One of the the following CPU capabilities: Primary, Quorum, Run, or Vector.
State	One of the following CPU states: Boot, Booted, Init, Rejected, Reserved, Run, Stopped, Stopping, or Timeout.
Mode	One of the following values for CPU modes supported for the architecture: Compatibility, Executive, Interrupt, Kernel, MP Synch, Null, Supervisor, or User. Note: Compatibility mode does not exist on OpenVMS Alpha systems.
% Used	A bar graph, by CPU, representing the percentage of the CPU utilization for each mode.
PID	The process identifier value of the process that is using the CPU. If the PID is unknown to the console application, the IPID will be listed.

(continued on next page)

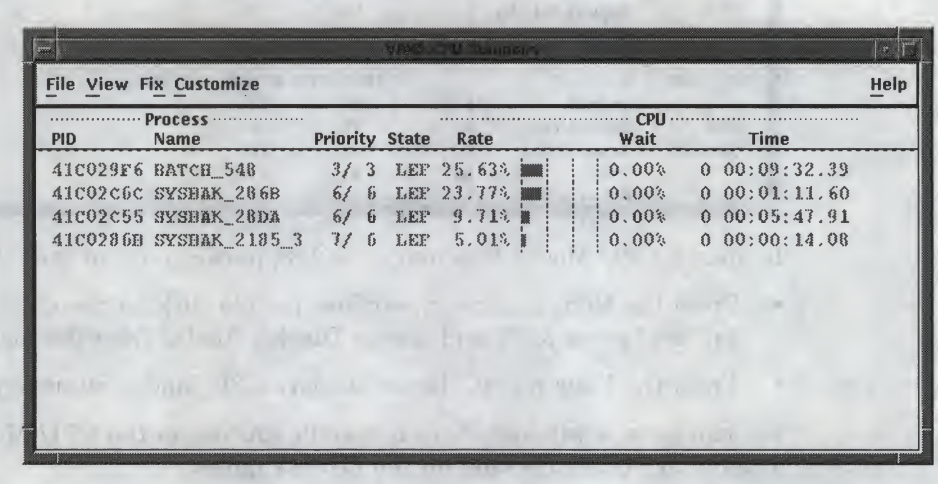
Table 3–2 (Cont.) CPU Modes Summary Window Data Fields

Field	Displays
Name	The process name of the process found in the CPU. If no process is found in the CPU, this will be listed as *** None ***.
Rate	A numerical percentage of CPU time for each mode.
Peak	The peak CPU usage determined for each mode.

3.2 CPU Summary Window

The CPU Summary window shown in Figure 3–3 displays summary statistics about process CPU usage issues that may be caused by CPU-intensive users or CPU bottlenecks.

Figure 3–3 CPU Summary Window



To open a CPU Summary window, perform one of the following steps:

- From the System Overview window, double click on the CPU field of any node. You can also press MB3 on an event that is related to CPU usage and choose Display from the menu.
- From the Node Summary window, double click on CPU Process State Queues.
- From the Event Log window, press MB3 on an event that is related to CPU usage and choose Display from the menu.

You can open a window about a specific process in the CPU Summary window by double clicking on the process name.

Table 3–3 describes the CPU Summary window data fields.

Table 3–3 CPU Summary Window Data Fields

Field	Displays
PID	The Process Identifier, a 32-bit value that uniquely identifies a process.
Name	The process name.
Priority	Computable *(xx)* and base *(yy)* process priority in the format *xx/yy*.
State	One of the values listed under the Single Process Summary description in Section 3.11.
Rate	The percent of CPU time used by this process. This is the ratio of CPU time to elapsed time. The CPU rate is also displayed in the bar graph.
Wait	The percent of time process is in the COM or COMO state.
Time	The amount of actual CPU time charged to the process.

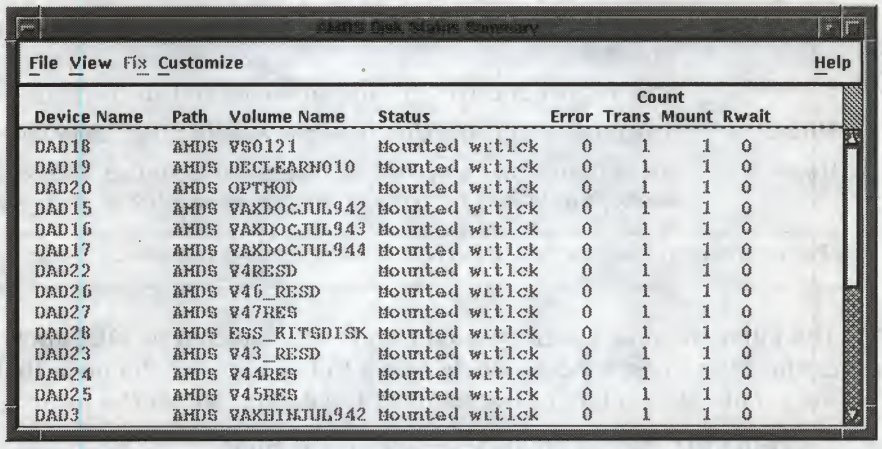
The following CPU-related events are detected by DECamds and displayed in the Event Log window. *Node* is replaced by the name of the node to which the event is related. *Process* is replaced by the name of the process to which the event is related.

PRCCOM, *Node Process* waiting in COM or COMO
PRCCVR, *Node Process* has high CPU rate
PRCMWT, *Node Process* waiting in MWAIT
PRCPWT, *Node Process* waiting in COLPG, PFW, or FPG

3.3 Disk Status Summary Window

The Disk Status Summary window (shown in Figure 3–4) displays data about availability, count, and errors of disk devices on the system.

Figure 3–4 Disk Status Summary Window



To open a Disk Status Summary window, perform one of the following steps:

- From the System Overview window, press MB3 on a node or group line, choose Display from the menu, and Disk Status Summary from the submenu.
- From the Event Log window, press MB3 on any Disk Status related event and choose Display from the menu.

Table 3–4 describes the Disk Status Summary window data fields.

Table 3–4 Disk Status Summary Window Data Fields

Field	Displays	
Device Name	The standard OpenVMS device name that indicates where the device is located, as well as a controller or unit designation.	
Path	The primary path (node) from which the device receives commands.	
Volume Name	The name of the media that is currently mounted.	
Status	One or more of the following disk status values:	
	Alloc	Disk is allocated to a specific user
	Clutran	Disk status is uncertain due to a cluster state transition in progress
	Dismount	Disk in process of dismounting; may be waiting for a file to close
	Foreign	Disk is mounted with the /FOREIGN qualifier
	Invalid	Disk is in an invalid state (likely Mount Verify Timeout)
	Mount Verify	Disk is waiting for a mount verification
	Mounted	Disk is logically mounted by a MOUNT command or service call
	Offline	Disk is no longer physically mounted in device drive
	Online	Disk is physically mounted in device drive
	Shadow Set Member	Disk is a member of a shadow set
	Unavailable	Disk is set /UNAVAILABLE
	Wrong Volume	Disk has been mounted with the wrong volume name
	Wrtlck	Disk is mounted and write locked
Errors[1]	The number of errors generated by the disk, (a quick indicator of device problems).	
Trans[1]	The number of currently-in-progress file system operations for the disk.	
Mount[1]	The number of nodes that have the specified disk mounted.	
Rwait[1]	An indicator that a system I/O operation is stalled, usually during normal connection failure recovery or volume processing of host-based shadowing.	

[1]For the group window, the sum of the node window values is displayed.

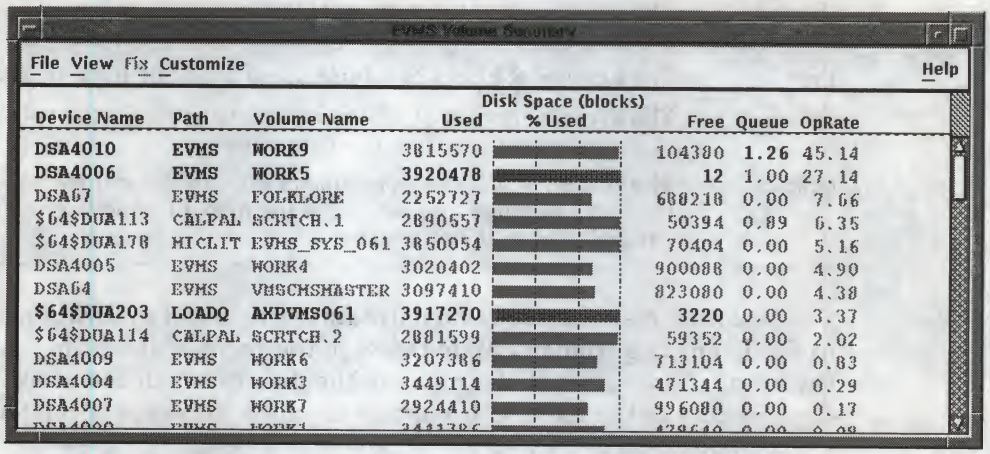
The following disk status-related events are detected by DECamds and displayed in the Event Log window. *Node* refers to the name of the node that is signalling the event. *Disk* refers to the name of the disk to which the event is related.

DSKERR, *Node Disk* disk error count is high
DSKINV, *Node Disk* disk is in an invalid state
DSKMNV, *Node Disk* disk mount verify in progress
DSKOFF, *Node Disk* disk is off line
DSKRWT, *Node Disk* disk Rwait count is high
DSKUNA, *Node Disk* disk is unavailable
DSKWRV, *Node Disk* wrong volume mounted

3.4 Disk Volume Summary Window

The Disk Volume Summary window shown in Figure 3–5 displays summary data about disk volumes mounted in the system. Disk volume summary data is accurate when every node in a VMScluster environment is in the same group. Multiple VMSclusters can share a group, but VMSclusters cannot be divided into different groups without losing accuracy.

Figure 3–5 Disk Volume Summary Window



Note

The group value for Free blocks used is determined from the node with the mastering lock on the volume resource.

To open a Disk Volume Summary window, perform one of the following steps:

- From the System Overview window, press MB3 on a node or group line, choose Display from the menu, and Disk Volume Summary from the submenu.
- From the Event Log window, press MB3 on any Disk Volume related event and choose Display from the menu.

Note

DECamds does not collect Volume Summary data on remote disks mounted using the VAX Distributed File Service (DFS).

Table 3–5 describes the Disk Volume Summary window data fields.

Table 3–5 Disk Volume Summary Window Data Fields

Field	Displays
Device Name	The standard OpenVMS device name that indicates where the device is located, as well as a controller or unit designation.
Path	The primary path (node) from which the device receives commands.
Volume Name	The name of the mounted media.
Used	The number of volume blocks in use.
% Used	The percentage of the number of volume blocks in use in relation to the total volume blocks available.
Free	The number of blocks of volume space available for new data.
Queue	The average number of I/O operations pending for the volume (an indicator of performance; less than one is optimal).
OpRate	The rate at which the Operations Count to the volume has changed since the last sampling, and measures the amount of activity on a volume; the optimal load is device specific.

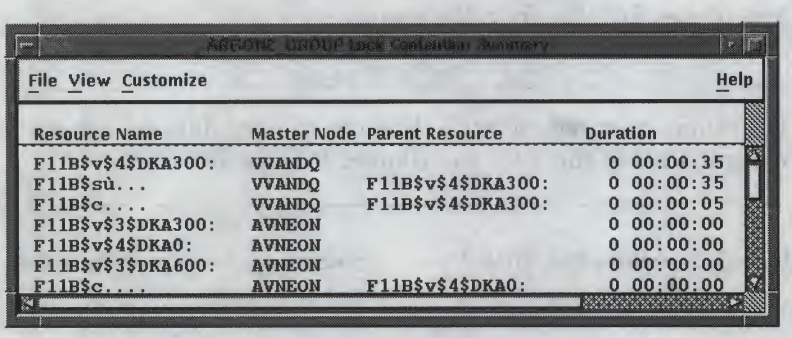
The following disk volume-related events are detected by DECamds and displayed in the Event Log window. *Node* refers to the name of the node that is signalling the event. *Disk* refers to the name of the disk to which the event is related. *Group* refers to the name of the group to which the event is related.

DSKQLN, *Node Disk* disk volume queue length is high
LOVLSP, *Group Disk* disk volume free space is low

3.5 Lock Contention Summary Window

The Lock Contention Summary window shown in Figure 3–6 is used to determine which resources are under contention. It displays all the OpenVMS Lock Manager resources that have potential lock contention situations. The Lock Contention Summary window is available only for groups; attempting to open a Lock Contention Summary for a node opens the node's group window.

Figure 3–6 Lock Contention Summary Window



Locks are written to AMDS$LOCK_LOG.LOG; see Section B.3 for more information. To interpret the information displayed in the Lock Contention Summary window, you should have an understanding of OpenVMS lock management services. For more information, see the *OpenVMS System Services Reference Manual*.

Note

Lock contention data is accurate only if every node in a VMScluster environment is in the same group. Multiple clusters can share a group, but VMSclusters cannot be divided into different groups without losing accuracy.

You can open a Lock Contention Summary window from the Event Log or System Overview windows, as follows:

- From the Event Log window, press MB3 on any lock contention-related event and choose Display from the menu.
- From the System Overview window:
  1. Press MB3 on any node or group line and choose Display from the menu.
  2. Choose Lock Contention Summary from the submenu.

Table 3–6 describes the Lock Contention Summary window data fields.

Table 3–6 Lock Contention Summary Window Data Fields

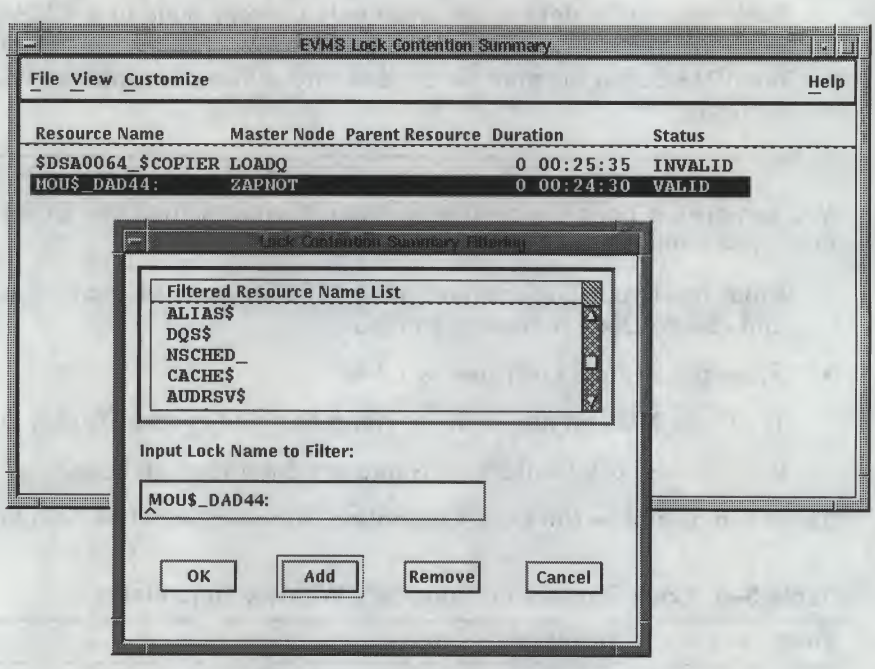
Field	Displays
Resource Name	The resource name associated with the $ENQ routine call.
Master Node	The node on which the resource is mastered.
Parent Resource Name	The name of the parent resource. If no name is displayed, the resource listed is the parent resource.
Duration	The amount of time elapsed since DECamds first detected the contention situation.
Status	The status of the lock. See the $ENQ(W) description in the *OpenVMS System Services Reference Manual*.

You can open a Single Lock Summary window from the Lock Contention Summary window. See Section 3.10 for more information.

Figure 3–7 shows how you can determine which filters can or cannot be displayed. To filter specific locks from the display, choose Filter Data... from the Customize menu on the Lock Contention Summary window. A filter dialog box appears with a list of locks currently being filtered from the display. You can add a filter by typing the name of a filter in the Input Lock Name to Filter field and clicking on the Add button. You can use the asterisk (*) wildcard character to specify a range of filters. For example, $DSA*$WAITER will filter all locks beginning with $DSA and ending with $WAITER and anything in between.

You can also click on an existing lock in the Lock Contention Summary window and it will be placed in the Input Lock Name to Filter field (as shown in Figure 3–7). You must click on the Add button to add the filter.

Figure 3–7 Filtering Lock Events



You can remove a lock from the filter list by selecting a lock and clicking on the Remove button. Any lock contentions affected by the removed filter will be displayed.

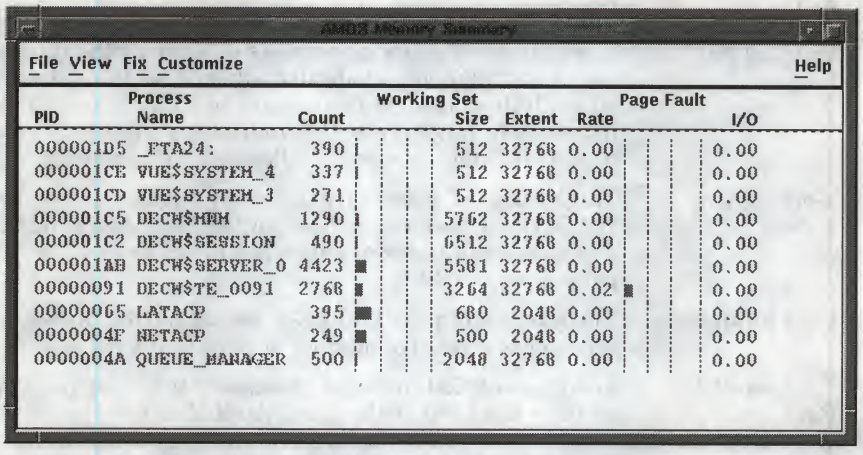
The following lock contention-related events are detected by DECamds and displayed in the Event Log window. Italicized words are replaced with actual values.

LCKCNT, *node* possible contention for resource *resource*
LRGHSH, *node* lock hash table too large *n* entries
RESDNS, *node resource* hash table dense *percentage* full *n* resources, hash table size *n*
RESPRS, *node* resource hash table sparse, only *percentage* full *n* resources, table size *n*

3.6 Memory Summary Window

The Memory Summary window shown in Figure 3–8 displays memory usage data for processes on a node so that you can identify processes that use large amounts of memory or have high page fault rates.

Figure 3–8 Memory Summary Window



To open a Memory Summary window, perform one of the following steps:

- From the Node Summary window, double click on the Page Faults or Memory area. You can also press MB3 on the Page Faults or Memory area and choose Display from the menu.
- From the View menu, choose Display Memory Summary.
- From the System Overview window, double click on the Memory field for any node. You can also press MB3 on any field for any node, choose Display from the pop-up menu, and Memory Summary from the submenu.
- To display a memory summary of every node in a group from the System Overview window, press MB3 on the group line, choose Display from the menu, and Memory Summary from the submenu.
- From the Event Log window, press MB3 on an event related to memory usage and choose Display from the menu.

You can open a window about a specific process in the Memory Summary window by double clicking on the process name.

Table 3–7 describes the Memory Summary window data fields.

Table 3–7 Memory Summary Window Data Fields

Field	Displays
PID	The process identifier, a 32-bit value that uniquely identifies a process.
Name	The process name.
Working Set Count[1]	The number of physical pages or pagelets of memory that the process is using. The bar graph represents the percentage of working set count used to the working set extent.

[1]Working Set Value = Total Physical Memory / Maximum Process Count

(continued on next page)

Table 3–7 (Cont.) Memory Summary Window Data Fields

Field	Displays
Working Set Size[1]	The number of pages or pagelets of memory the process is allowed to use. This value is periodically adjusted by the operating system based on analysis of page faults relative to CPU time used. When the value increases in large units, this indicates a process is receiving a lot of page faults and its memory allocation is increasing.
Working Set Extent[1]	The number of pages or pagelets of memory in the process's WSEXTENT quota as defined in the User Authorization File (UAF). The number of pages or pagelets will not exceed the value of the system parameter WSMAX.
Page Fault Rate	The number of page faults per second for the process. The bar graph represents a relative number of page faults per second.
Page Fault I/O Rate	The rate of READ attempts necessary to satisfy page faults (also known as Page Read I/O or the Hard Fault Rate).

[1]Working Set Value = Total Physical Memory / Maximum Process Count

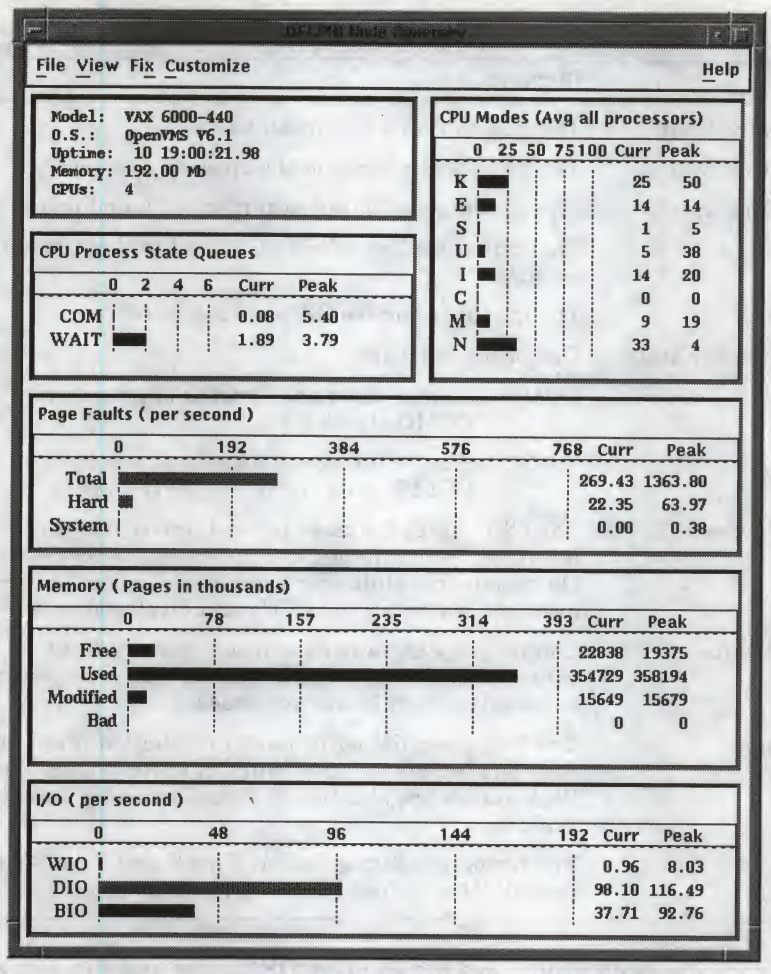
The following memory-related events are detected by DECamds and displayed in the Event Log window. *Node process* is replaced by the name of the process to which the event is related.

LOWEXT, *node process* working set extent is too small
LOWSQU, *node process* working set quota is too small
PRPGFL, *node process* high page fault rate
PRPIOR, *node process* paging I/O rate is high

3.7 Node Summary Window

The Node Summary window shown in Figure 3–9 displays a high-level graphic summary of node resource demands on the CPU, Memory, and I/O.

Figure 3–9 Node Summary Window



To open a Node Summary window, perform one of the following steps:

- From the System Overview window, double click on any node name. You can also press MB3 on any node name and choose Display from the menu.
- From the Event Log window, double click on any node name. You can also press MB3 on an event that is related to node summary data and choose Display from the menu.

Dynamic bar graphs display the current values for each field. Peak values are also displayed from when DECamds begins collecting node summary data. A peak value is typically the highest value received; however, for the Free Memory field it's the lowest value seen.

You can open the following windows from the Node Summary Window by double clicking in the space for each category:

CPU Summary
CPU Modes Summary
Memory Summary
Process I/O summary

Table 3–8 describes the Node Summary window data fields.

Table 3–8 Node Summary Window Data Fields

Field	Displays
Hardware Model	The system hardware model name.
Operating System	The name and version of the operating system.
Total Memory	The total amount of physical memory found on the system.
Uptime	The time since last reboot measured in days, hours, minutes, and seconds.
Active	The number of active CPUs on the node.
CPU Process State Queues	One of the following: *COM* Sum of the queue lengths of processes in the COM and COMO states. *WAIT* Sum of the queue lengths of processes in the MWAIT, COLPG, CEF, PFW, and FPG states.
CPU Modes	The CPU usage by mode (kernel, executive, supervisor, user, interrupt, compatibility, multiprocessor synchronization, and null). On Symmetric Multiprocessing (SMP) nodes, percentages are averaged across all the CPUs and displayed as one value.
Page Faults	The rate of system hard and soft page faulting, as well as peak values seen during a DECamds session. System page faults are those taken from kernel processes.
Memory	The histogram listing memory distribution (Free, In Use, Modified, Bad) as absolute values of number of thousands of pages or pagelets. Peak values are also listed, with Free using lowest seen value as peak.
I/O	The histogram listing Buffer, Direct, and Page Write I/O rates per second. Also included is the peak value seen.

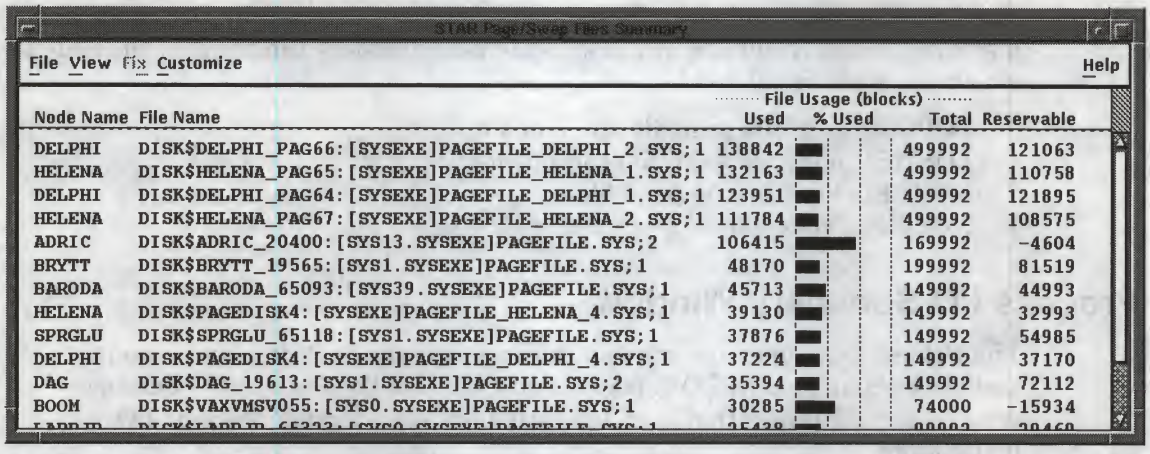
The following node events are detected by DECamds and displayed in the Event Log window. *Node* is replaced by the name of the node to which the event is related.

HIBIOR, *node* buffered I/O rate is high
HICOMQ, *node* many processes waiting for CPU
HIDIOR, *node* direct I/O rate is high
HIHRDP, *node* hard page fault rate is high
HIMWTQ, *node* process waiting in MWAIT
HINTER, *node* interrupt mode time is high
HIPWIO, *node* paging write I/O rate is high
HIPWTQ, *node* many processes waiting in COLPG, PFW, or FPG
HITTLP, *node* total page fault rate is high
HMPSYN, *node* MP synchronization mode time is high
HISYSP, *node* system page fault rate is high
LOMEMY, *node* free memory is low
NOPROC, *node* cannot find process names *process*

3.8 Page/Swap File Summary Window

The Page/Swap File Summary window shown in Figure 3–10 displays data about a node's page/swap file usage and is used to identify page or swap files that are overused or underconfigured. It is also used to find nodes that lack a page or swap file.

Figure 3–10 Page/Swap File Summary Window



You can open a Page/Swap File Summary window from the Event Log or System Overview windows, as follows:

- From the Event Log window, press MB3 on any Page/Swap space low-related event and choose Display from the menu.
- From the System Overview window:
  1. Press MB3 on any node or group line and choose Display from the menu.
  2. Choose Page/Swap File Summary from the submenu.

Table 3–9 describes the Page/Swap File Summary window data fields.

Table 3–9 Page/Swap File Summary Window Data Fields

Field	Displays
Node Name	The name of the node on which the page/swap file resides.
File Name	The name of the page/swap file. For secondary page/swap files, the file name is obtained by a special AST to the job controller on the remote node. DECamds makes one attempt to retrieve the file name.
Used	The number of used pages or pagelet blocks within the file.
% Used	A graph representing the percentage of the blocks from the available page or pagelet blocks in each file.
Total	The total number of pages or pagelet blocks within the file.

(continued on next page)

Table 3–9 (Cont.) Page/Swap File Summary Window Data Fields

Field	Displays
Reservable	The number of pages or pagelet blocks that can be logically claimed by a process for future physical allocation. This value can be listed as a negative value, since it is merely a value of a process's interest in getting pages from the file. If every process currently executing needed to use the file, then this value is the debt that is owed.

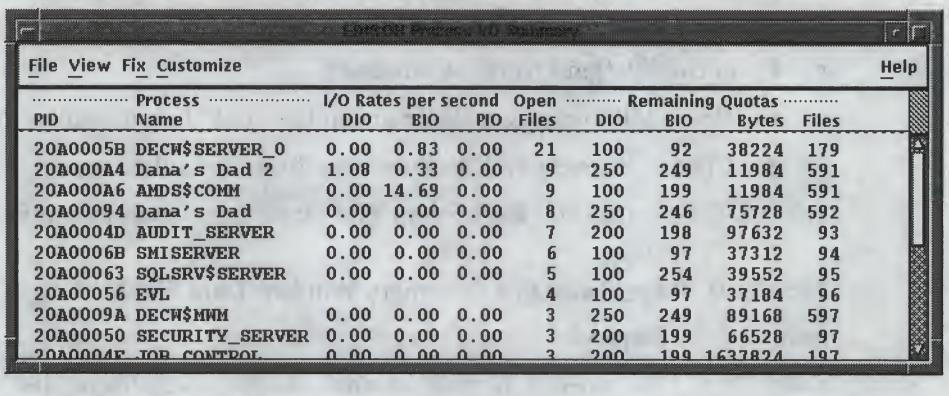
The following page and swap file-related events are detected by DECamds and displayed in the Event Log window. *Node* is replaced by the name of the node to which the event is related.

LOPGSP, *node* file page file space is low
LOSWSP, *node* file swap file space is low
NOPGFL, *node* has no page file
NOSWFL, *node* has no swap file

3.9 Process I/O Summary Window

The Process I/O Summary window shown in Figure 3–11 displays summary statistics about process I/O rates and quotas. Use the Process I/O Summary window for I/O issues that may be caused by I/O-intensive users or I/O bottlenecks.

Figure 3–11 Process I/O Summary Window



To open a Process I/O Summary window, perform one of the following steps:

- From the Node Summary window, double click in the I/O area.
- From the View menu, choose Display Process I/O Summary.
- From the System Overview window, double click on the BIO or DIO fields for any node. You can also press MB3 on any field for any node, choose Display from the menu, and Process I/O Summary from the submenu.
- To open a Process I/O Summary window for every node in a group, from the System Overview window, press MB3 on a group line, choose Display from the menu, and Process I/O Summary from the submenu.
- From the Event Log window, press MB3 on any process I/O related event and choose Display from the menu.

You can open a window about a specific process in the Process I/O Summary window by double clicking on the process name.

Table 3–10 describes the Process I/O Summary window data fields.

Table 3–10 Process I/O Summary Window Data Fields

Field	Displays
PID	The process identifier, a 32-bit value that uniquely identifies a process.
Process Name	The current process name.
Direct I/O Rate (DIO)	The rate at which I/O transfers occur between the system devices and the pages or pagelets that contain the process buffer that the system locks in physical memory.
Buffered I/O Rate (BIO)	The rate at which I/O transfers occur between the process buffer and an intermediate buffer from the system buffer pool.
Paging I/O Rate (PIO)	The rate of READ attempts necessary to satisfy page faults (also known as Page Read I/O or the Hard Fault Rate).
Open Files	The number of open files.
Direct I/O Limit Remaining (DIO)	The number of remaining direct I/O limit operations available before the process reaches its quota. DIOLM quota is the maximum number of direct I/O operations a process may have outstanding at one time.
Buffered I/O Limit Remaining (BIO)	The number of remaining buffered I/O operations available before the process reaches its quota. BIOLM quota is the maximum number of buffered I/O operations a process may have outstanding at one time.
Byte Limit Remaining (Bytes)	The number of buffered I/O bytes available before the process reaches its quota. BYTLM is the maximum number of bytes of nonpaged system dynamic memory that a process can claim at one time.
Open File Limit Remaining (Files)	The number of additional files the process can open before reaching its quota. FILLM quota is the maximum number of files that can be opened simultaneously by the process, including active network logical links.

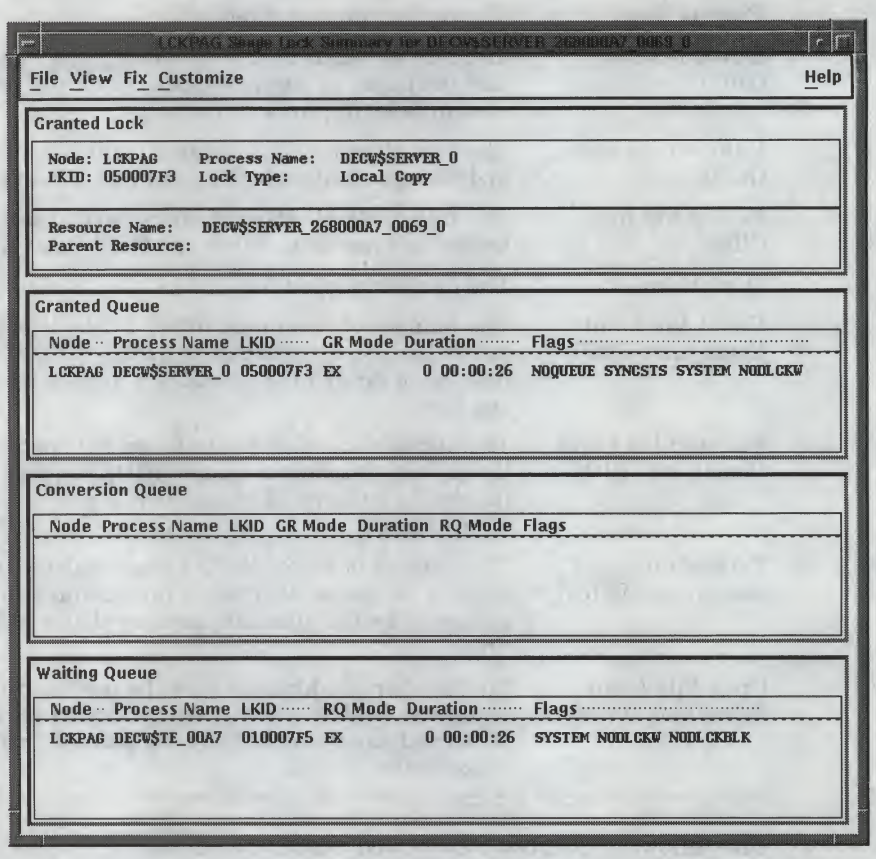
The following process I/O-related events are detected by DECamds and displayed in the Event Log window. *Node process* is replaced by the name of the process to which the event is related.

LOBIOQ, *node process* has used most of its BIOLM process quota
LOBYTQ, *node process* has used most of its BYTLM job quota
LODIOQ, *node process* has used most of its DIOLM process quota
LOFILQ, *node process* has used most of its FILLM job quota
PRBIOR, *node process* buffered I/O rate is high
PRDIOR, *node process* direct I/O rate is high
PRPIOR, *node process* paging I/O rate is high

3.10 Single Lock Summary Window

The Single Lock Summary window shown in Figure 3–12 displays data about a blocking lock and all locks in the granted, conversion, and waiting queues. Use it to get detailed information about a lock contention situation. The lock name is specified in the title bar. All locks are written to AMDS$LOCK_LOG.LOG; see Section B.3 for more information.

Figure 3–12 Single Lock Summary Window



If a Single Lock Summary window cannot determine the node name for the group it uses the Cluster System Identification (CSID) value, which is used by the VMScluster software to uniquely identify cluster members.

To open a Single Lock Summary window, perform one of the following steps:

- From the Lock Contention Summary window, double click on any field. You can also press MB3 on any field and choose Display Lock from the menu.
- From the View menu, choose Display Lock.
- From the Event Log window, press MB3 on any lock blocking-related or lock waiting-related event and choose Display from the menu.

Table 3–11 describes the Single Lock Summary window data fields.

Table 3–11 Single Lock Summary Window Data Fields

Field	Displays
	Granted Lock
Type	One of the following: Local Copy, Process Copy, or Master Copy.
Node	The node name on which the lock is granted.
Process Name	The name of the process owning the blocking lock.
LKID	The lock ID value (which is useful with SDA).
Resource Name	The name of the resource.
Parent Resource Name	The name of the parent resource (if any).
	Granted, Conversion, and Waiting Queue
Node	The node on which the lock block resides.
Process Name	The process name of the process owning the lock.
LKID	The Lock ID value (which is useful with SDA).
GR Mode	One of the following modes at which the lock is granted: EX, CW, CR, PW, PR, NL.
RQ Mode	One of the following modes at which the lock is requested: EX, CW, CR, PW, PR, NL.
Duration	The length of time lock has been in current queue with respect to time the console application has found it.
Flags	The flags specified with the $ENQ(W) request.

You can open a window about a specific process in the Single Lock Summary window by double clicking on the process name.

Note

Processes that are labeled *unknown* are associated with system locks. They cannot be opened.

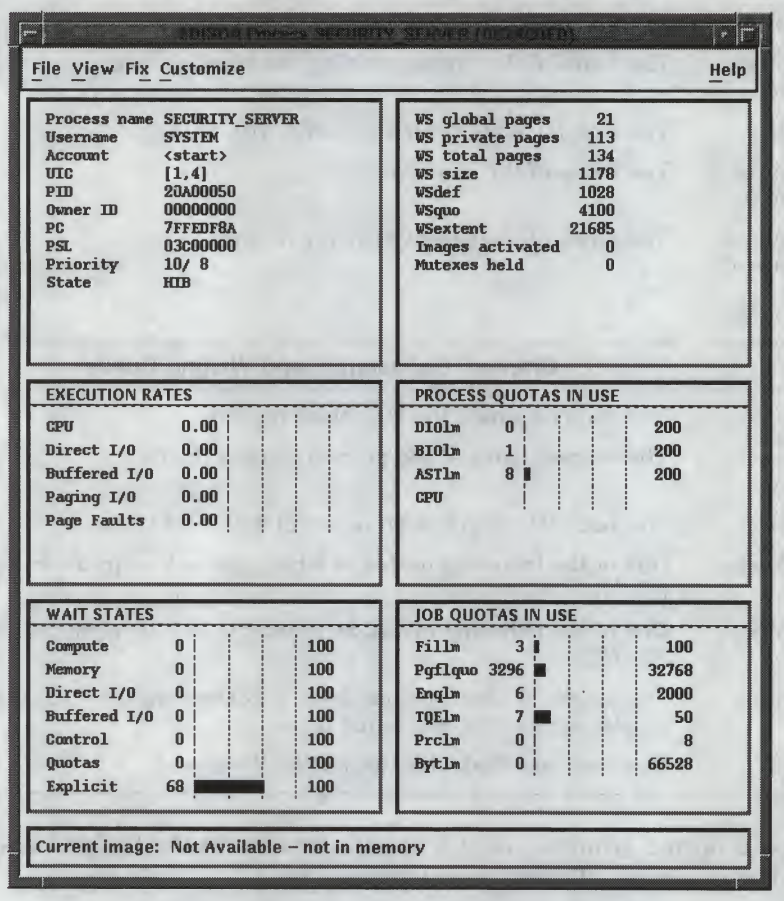
The following single lock-related events are detected by DECamds and displayed in the Event Log window. Italicized words are replaced with actual values.

LCKBLK, *node process* blocking resource *resource*
LCKWAT, *node process* waiting for resource *resource* granted to process on node *node*

3.11 Single Process Summary Window

The Single Process Summary window shown in Figure 3–13 displays summary data about a process, including Execution Rates, Process Quotas in Use, Wait States, and Job Quotas in Use.

Figure 3–13 Single Process Summary Window



To open a Single Process Summary window, perform one of the following steps:

- From any window that displays processes (CPU, CPU Modes, Memory, Process I/O, and Single Lock Summary), double click on any field. You can also press MB3 on any field in a process line and choose Display from the pop-up menu.
- You can also click on any field in a process line and choose Display Process from the View menu.
- From the Event Log window, double click on a process-related event. You can also press MB3 on a process-related event, choose Display from the menu, and Single Process in the dialog box.

Table 3–12 describes the Single Process Summary window data fields.

Table 3–12 Single Process Summary Window Data Fields

Field	Displays
Process Name	The name of the process.
Username	The username of the user owning the process.
Account	The string assigned to the user by the system manager.
UIC	The user identification code (UIC), a pair of numbers or character strings designating the group and user.
PID	The process identifier, a 32-bit value that uniquely identifies a process.
Owner ID	The process identifier (PID) of the process that created the process displayed in the window. If 0, then the process is a parent process.
PC	The program counter. On OpenVMS VAX systems, this is the address of the next instruction the CPU will execute. On OpenVMS Alpha systems, this value is displayed as 0, since the data is not readily available to the Data Provider node.
PSL	The Processor Status Longword (PSL). On OpenVMS VAX systems, this indicates the current processor mode (user, kernel, and so on) and its interrupt level. On OpenVMS Alpha systems, this value is displayed as 0, since the data is not readily available to the Data Provider node.
Priority	The computable and base priority of the process. Priority is an integer between 0 and 31. Processes with higher priority get more CPU time.

(continued on next page)

Table 3–12 (Cont.) Single Process Summary Window Data Fields

Field	Displays	
State	One of the following process states:	
	CEF	Common Event Flag, waiting for a Common Event Flag
	COLPG	Collided Page Wait, involuntary wait state; likely indicates a memory shortage, waiting for hard page faults
	COM	Computable; ready to execute
	COMO	Computable Outswapped, COM, but swapped out
	CUR	Current, currently executing in a CPU
	FPW	Free Page Wait, involuntary wait state; likely indicates a memory shortage
	LEF	Local Event Flag, waiting for a Local Event Flag
	LEFO	Local Event Flag Outswapped; LEF, but outswapped
	HIB	Hibernate, voluntary wait state requested by the process; it is inactive
	HIBO	Hibernate Outswapped, hibernating but swapped out
	MWAIT	Miscellaneous Resource Wait, involuntary wait state; possibly caused by a shortage of a systemwide resource such as no page or swap file capacity or synchronizations for single threaded code
	PFW	Page Fault Wait, involuntary wait state; possibly indicates a memory shortage, waiting for hard page faults
	RWAST	Resource Wait State, waiting for delivery of an asynchronous system trap (AST) that signals a resource availability; usually an I/O is outstanding or a process quota is exhausted
	RWBRK	Resource Wait for BROADCAST to finish
	RWCAP	Resource Wait for CPU Capability
	RWCLU	Resource Wait for Cluster Transition
	RWCSV	Resource Wait for Cluster Server Process
	RWIMG	Resource Wait for Image Activation Lock
	RWLCK	Resource Wait for Lock ID data base
	RWMBX	Resource Wait on MailBox, either waiting for data in mailbox (to read) or waiting to place data (write) into a full mailbox (some other process has not read from it; mailbox is full so this process cannot write).
	RWMPB	Resource Wait for Modified Page writer Busy
	RWMPE	Resource Wait for Modified Page list Empty
	RWNPG	Resource Wait for Non Paged Pool
	RWPAG	Resource Wait for Paged Pool
	RWPFF	Resource Wait for Page File Full
	RWQUO	Resource Wait for Pooled Quota
	RWSCS	Resource Wait for System Communication Services
	RWSWP	Resource Wait for Swap File space
	SUSP	Suspended, wait state process placed into suspension; it can be resumed at the request of an external process
	SUSPO	Suspended Outswapped, suspended but swapped out

(continued on next page)

Table 3–12 (Cont.) Single Process Summary Window Data Fields

Field	Displays
WS Global Pages	The shared data or code between processes listed in pages or pagelets.
WS Private Pages	The amount of accessible memory, listed in pages or pagelets.
WS Total Pages	The sum of global and private pages or pagelets.
WS Size	The working set size, number of pages or pagelets of memory the process is allowed to use. This value is periodically adjusted by the operating system based on analysis of page faults relative to CPU time used. When it increases in large units, this indicates a process is taking a lot of page faults and its memory allocation is increasing.
WSdef	The working set default, the initial limit to the number of physical pages or pagelets of memory the process can use. This parameter is listed in the user authorization file (UAF); discrepancies between the UAF value and the displayed value are due to page/longword boundary rounding or other adjustments made by the operating system.
WSquo	The working set quota, the maximum amount of physical pages or pagelets of memory the process can lock into its working set. This parameter is listed in the UAF; discrepancies between the UAF value and the displayed value are due to page/longword boundary rounding or other adjustments made by the operating system.
WSextent	The working set extent, the maximum number of physical pages or pagelets of memory the system will allocate for the process. The system provides memory to a process beyond its quota only when it has an excess of free pages and can be recalled if necessary. This parameter is listed in the UAF; any discrepancies between the UAF value and the displayed value are due to page/longword boundary rounding or other adjustments made by the operating system.
Images Activated	The number of times an image is activated.
Mutexes Held	The number of mutual exclusions (mutexes) held. Persistent values other than zero (0) require analysis. A mutex is similar to a lock but is restricted to one CPU. When a process holds a mutex, its priority is temporarily incremented to 16.

(continued on next page)

Table 3–12 (Cont.) Single Process Summary Window Data Fields

Field	Displays
	Execution Rates
CPU Rate	The percent of CPU time used by this process. This is the ratio of CPU time to elapsed time. CPU rate is also displayed in the bar graph.
Direct I/O Rate (DIO)	The rate at which I/O transfers take place from the pages or pagelets containing the process buffer that the system locks in physical memory to the system devices.
Buffered I/O Rate (BIO)	The rate at which I/O transfers take place for the process buffer from an intermediate buffer from the system buffer pool.
Paging I/O Rate (PIO)	The rate of READ attempts necessary to satisfy page faults. This is also known as Page Read I/O or the Hard Fault Rate.
Page Fault Rate	The page faults per second for the process. The bar graph visually represents page faults per second.
	Process Quotas in Use[1]
Direct I/O Limit (DIOLM)	A bar graph representing current count of DIOs used with respect to the limit that can be attained.
Buffered I/O Limit (BIOLM)	A bar graph representing current count of BIOs used with respect to the limit that can be attained.
Asynchronous System Traps Limit (ASTLM)	A bar graph representing current count of ASTs used with respect to the limit that can be attained.
CPU Time Limit (CPULM)	A bar graph representing current count of CPU Time used with respect to the limit that can be attained. If the limit is 0, then this value is not used.
	Wait States[2]
Compute	A relative value indicating that the process is waiting for CPU time. The included states are COM, COMO, RWCAP.
Memory	A relative value indicating that the process is waiting for a page fault that requires data to be read from disk; common during image activation. The included states are PFW, COLPG, FPG, RWPAG, RWNPG, RWMPE, RWMPB.
Direct I/O	A relative value indicating that the process is waiting for data to be read from or written to a disk. The included state is DIO.
Buffered I/O	A relative value indicating that the process is waiting for data to be read from or written to a slower device such as a terminal, line printer, or mailbox. The included state is BIO.
Control	A relative value indicating that the process is waiting for another process to release control of some resource. The included states are CEF, MWAIT, LEF, LEFO, RWAST, RWMBX, RWSCS, RWCLU, RWCSV, RWUNK, and LEF waiting for a ENQ.
Quotas	A relative value indicating that the process is waiting because the process has exceeded some quota. The included states are QUOTA and RWAST_QUOTA.

[1]When you display the SWAPPER process, no values are listed in this section. The SWAPPER process does not have quotas defined in the same way other system and user processes do.

[2]The wait state specifies why a process cannot execute, based on application-specific calculations.

(continued on next page)

Table 3–12 (Cont.) Single Process Summary Window Data Fields

Field	Displays
	Wait States[2]
Explicit	A relative value indicating that the process is waiting because process asked to wait, such as a hibernate system service. The included states are HIB, HIBO, SUSP, SUSPO, and LEF waiting for a TQE.
	Job Quotas in Use
File Limit (FILLM)	A bar graph representing current number of open files with respect to the limit that can be attained.
Page File Quota (PGFLQUO)	A bar graph representing current number of disk blocks in page file that the process can use with respect to the limit that can be attained.
Enqueue Limit (ENQLM)	A bar graph representing current count of resources (lock blocks) queued with respect to the limit that can be attained.
Timer Queue Entry Limit (TQELM)	A bar graph representing current count of timer requests with respect to the limit that can be attained.
Process Limit (PRCLM)	A bar graph representing current count of subprocesses created with respect to the limit that can be attained.
Buffered I/O Byte Limit (BYTLM)	A bar graph representing current count of bytes used for buffered I/O transfers with respect to the limit that can be attained.
Image Name	The name of the currently executing image - if available. If this field does not appear, then the data is not resident in memory.

[2]The wait state specifies why a process cannot execute, based on application-specific calculations.

The following single process-related events are detected by DECamds and displayed in the Event Log window. *Node* is replaced by the name of the node to which the event is related. *Process* is replaced by the name of the process to which the event is related.

LOASTQ, *node process* has used most of its ASTLM process quota
LOBIOQ, *node process* has used most of its BIOLM process quota
LOBYTQ, *node process* has used most of its BYTLM job quota
LODIOQ, *node process* has used most of its DIOLM process quota
LOENQU, *node process* has used most of its ENQLM job quota
LOFILQ, *node process* has used most of its FILLM job quota
LOPGFQ, *node process* has used most of its PGFLQUOTA job quota
LOPRCQ, *node process* has used most of its PRCLM process quota
LOTQEQ, *node process* has used most of its TQELM job quota
LOWEXT, *node process* working set extent is too small
LOWSQU, *node process* working set quota is too small
PRBIOR, *node process* buffered I/O rate is high
PRBIOW, *node process* waiting for buffered I/O
PRCCOM, *node process* waiting in COM or COMO
PRCCUR, *node process* has high CPU rate
PRCMUT, *node process* waiting for a mutex
PRCPUL, *node process* has used most of its CPULIM process quota
PRCPWT, *node process* waiting in COLPG, PFW, or FPG
PRCQUO, *node process* waiting for a quota
PRCRWA, *node process* waiting in RWAST
PRCRWC, *node process* waiting in RWCAP

PRCRWM, *node process* waiting in RWMBX
PRCRWP, *node process* waiting in RWPAG, PWNPG, RWMPE, or RWMPB
PRCRWS, *node process* waiting in RWSCS, RWCLU, or RWCSV
PRCUNK, *node process* waiting for a system resource
PRDIOR, *node process* direct I/O rate is high
PRDIOW, *node process* waiting for direct I/O
PRLCKW, *node process* waiting for a lock
PRPGFL, *node process* high page fault rate
PRPIOR, *node process* paging I/O rate is high

4

Performing Fixes

You can perform **fixes** to resolve resource availability problems and improve system availability. However, performing certain actions to fix a problem can have serious repercussions on a system including possibly causing a system failure. Therefore, only experienced system managers should perform fixes.

This chapter covers the following topics:

- Understanding fixes
- Performing fixes
- Typical fix examples

4.1 Understanding Fixes

When DECamds detects a resource availability problem, it analyzes the problem and proposes one or more fixes to improve the situation. Most fixes correspond to an OpenVMS system service call.

The following fixes are available from DECamds:

Fix Category	Possible Fixes	System Service Call
Memory usage fixes	Adjust working set size Purge working set	$ADJWSL $PURGWS
Process fixes	Delete a process Exit an image	$DELPRC $FORCEX
Process Quota Limit Adjust	Change limits for AST, BIO, DIO, ENQ, FIL, PRC, and TQE process quota limits	none
Process state fixes	Resume a process Suspend a process	$RESUME $SUSPND
Process priority fixes	Lower or raise a process priority	$SETPRI
Quorum fix	Adjust cluster quorum	None
System fix	Crash node	None

Before you perform a fix, you should understand the following information:

- Fixes are optional.
- You must have write access to perform a fix. (See Section 1.3 for more information about DECamds security.)
- You cannot undo many fixes. (After using the crash node fix, for example, the node must be rebooted.)
- The exit image, delete process, and suspend process fixes should not be applied to system processes. Doing so can also require rebooting the node.

- Whenever you exit an image, you cannot return to that image.
- Processes that have exceeded their job or process quota cannot be deleted.
- DECamds ignores fixes applied to the SWAPPER process.

4.2 Performing Fixes

Standard OpenVMS privileges restrict write access of users. When you run the Data Analyzer, you must have the CMKRNL privilege to send a write (fix) instruction to a node with a problem.

To initiate a fix, perform one of the following actions:

- From any of the data windows, double click on a process, then choose an action from the Fix menu.
- Press MB3 on an event and choose Fix from the menu.

DECamds displays a dialog box listing the fixes you can perform for the selected event. The recommended choice is highlighted. When you click on OK or Apply, DECamds performs one of the following actions:

- If the event you selected is not specific to a certain process, DECamds automatically performs the fix. Some fixes are performed automatically when "(automatic)" is displayed next to the selection.
- If the event is specific to a process, DECamds displays another dialog box in which you can specify the fix parameters. For example, for the Adjust Working Set Size fix, you specify a new working set size for the process.

  DECamds performs the highlighted fix as long as the event still exists. If the event you are fixing has changed, the dialog box disappears when you click on OK, Apply, or Cancel, and the fix is not performed.

Table 4–1 summarizes all fixes along with the windows from which they are available.

Table 4–1 Summary of DECamds Fixes

Problem to be Solved	Fix	Available From	Effects
Cluster hung	Adjust Quorum	Node Summary	Quorum for cluster is adjusted
Node or process low memory	Purge Working Set	Event Log Memory Summary Single Process Summary	Free memory, initial increment to page fault rate
Node resource hanging cluster	Crash Node	System Overview Node Summary Single Lock Summary	Node crashes with operator requested shutdown
Process looping, intruder	Delete Process	Any process window	Process no longer exists
Process loop in same PC range endlessly	Exit Image Process	Any process window	Exit from current image

(continued on next page)

Table 4–1 (Cont.) Summary of DECamds Fixes

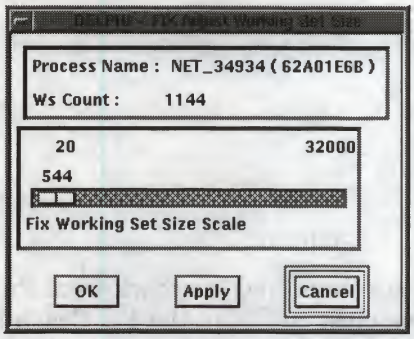
Problem to be Solved	Fix	Available From	Effects
Process previously suspended	Resume Process	Event Log Memory Summary CPU Summary Process I/O Summary Single Process Summary	Process will start from point it was suspended
Process quota has reached its limit and has entered RWAIT state	Adjust Process Quota Limit	Single Process Summary Event Log	Process receives greater limit
Runaway process	Change Process Priority	CPU Summary Single Process Summary Event Log	Priority stays at selected setting
Runaway process, unwelcome intruder	Suspend Process	Event Log Memory Summary CPU Summary Process I/O Summary Single Process Summary	Process gets no computes
Working set too high or low	Adjust Working Set Size	Memory Summary Single Process Summary Event Log	Page faulting might occur, might take up too much memory

The following sections in this chapter provide reference information about each DECamds fix.

4.2.1 Adjust Working Set Fix

When you perform the Adjust Working Set fix, DECamds displays a dialog box similar to the one shown in Figure 4–1.

Figure 4–1 FIX Adjust Working Set Size Dialog Box



Adjusting the working set can give needed memory to other processes that are page faulting. The adjustment made should attempt to bring the working set size closer to the actual count being used by non-page faulting processes.

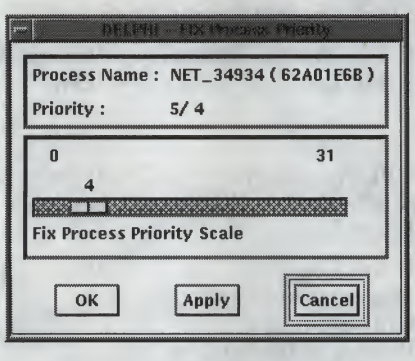
Caution

If the automatic working set adjustment is enabled for the system, a fix to Adjust Working Set will disable the automatic adjustment for the process.

4.2.2 Change Process Priority Fix

When you perform the Change Process Priority fix, DECamds displays a dialog box similar to Figure 4–2.

Figure 4–2 FIX Process Priority Dialog Box

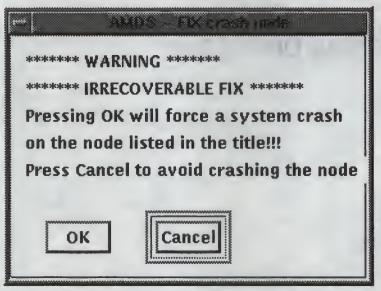


Setting a priority too high for a compute-bound process allows it to consume all the CPU cycles on the node, which can affect performance dramatically. On the other hand, setting a priority too low prevents the process from getting enough CPU cycles to do its job, which can also affect performance.

4.2.3 Crash Node Fix

When you perform the Crash Node fix, DECamds displays a dialog box similar to Figure 4–3.

Figure 4–3 FIX Crash Node Dialog Box



Caution

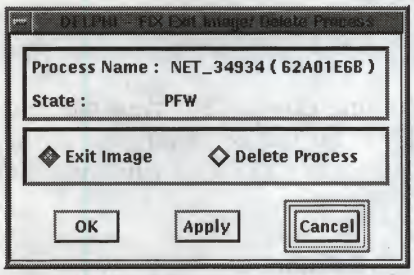
The crash node fix is an operator-requested bugcheck from the driver. It happens immediately when you click on OK in the Fix Crash Node dialog box. After performing this fix, the node cannot be restored to its previous state. After a crash, the node must be rebooted.

4.2.4 Exit Image and Delete Process Fixes

This section describes the Process State fixes.

When you perform either the Exit Image or Delete Process fix, DECamds displays a dialog box similar to Figure 4–4.

Figure 4–4 FIX Process State Dialog Box — Exit Image or Delete Process



You cannot reverse the action when you delete a process that is in a resource wait state. You must reboot the node. Deleting a process on a node that is in a resource wait state might not have an effect on the process.

Exiting an image on a node can stop an application that is required by the user. Check the single process window first to determine what image it is running.

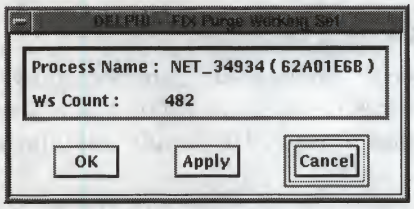
Caution

Deleting or exiting a system process could corrupt the kernel.

4.2.5 Purge Working Set Fix

When you perform the Purge Working Set fix, DECamds displays a dialog box similar to Figure 4–5.

Figure 4–5 FIX Purge Working Set Dialog Box

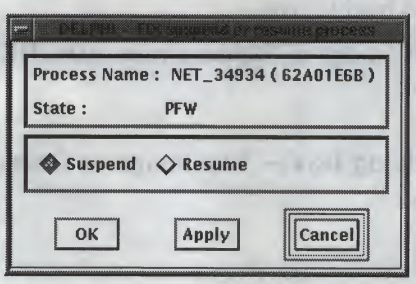


Continual purging of a working set on a node could force excessive page faulting, which affects system performance.

4.2.6 Suspend Process and Resume Process Fixes

When you perform either the Suspend Process or Resume Process fix, DECamds displays a dialog box similar to the one shown in Figure 4–6.

Figure 4–6 FIX Process State Dialog Box — Suspend or Resume Process



Suspending a process that is consuming excess CPU time can improve perceived CPU performance by freeing the CPU for use by other processes. Conversely, resuming a process that was using excess CPU time while running, may reduce perceived CPU performance.

Caution

Do not suspend system processes, especially JOB_CONTROL.

4.3 Examples for Fixing Low Memory Availability

This section describes two approaches for solving a low memory problem, which is a common resource availability problem.

The first example uses DECamds default settings. The second example shows how you can use DECamds to make a more detailed analysis and investigation. Both examples begin at the Event Log window entry.

4.3.1 Performing a Fix Using Automatic Fix Settings

When a process is page faulting, for example, it may signal a problem of available memory. A low memory (LOMEMY) event is generated. To fix this problem, you should purge the working sets of inactive processes. This will free up memory for the process that is page faulting. DECamds offers a quick, direct way to fix this and similar problems by performing the following steps:

1. Press MB3 on the event and choose Fix.

   If the event is related to a specific process, DECamds displays a dialog box with fixes you can perform. If the event is not related to a specific process but may be related to more than one process, DECamds automatically performs the fix.

   In the low memory example, DECamds displays a dialog box suggesting the automatic Purge Working Set fix.

2. Click on OK or Apply to perform the fix.

   The Purge Working Set fix purges the working set of the five processes that are the highest consumers of memory and are not page faulting. If this fix is not sufficient and the low memory event entry returns, repeat the fix every 15 or 20 seconds until enough working sets are purged to eliminate the event message. If two or three purges are not sufficient, then you should investigate manually.

4.3.2 Performing a Fix Using Manual Investigation

DECamds lets you manually display additional information related to an event before performing a fix. The following example uses the same low memory problem described in the previous section to investigate and select specific fixes for the problem.

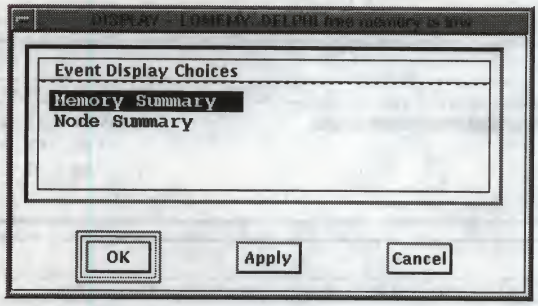
For this example, manually select the processes you want to fix from the Memory Summary window. You also may want to refer to data in the CPU Summary window.

To investigate the low memory event before fixing it, perform the following steps:

1. Press MB3 on the event to display the menu and choose Display.

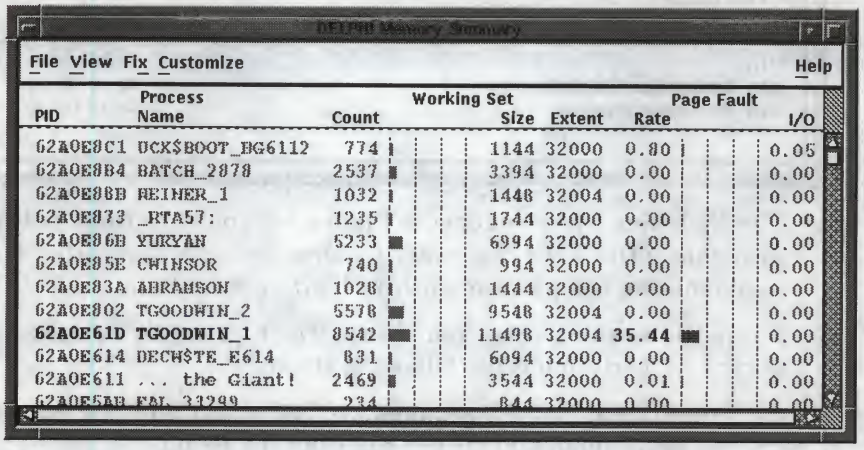
   DECamds displays a dialog box with a window name highlighted to indicate the recommended path for information. In the example shown in Figure 4–7, the Memory Summary window is recommended.

Figure 4–7 Sample Fix Dialog Box



2. Click on Apply to open the Memory Summary window (Figure 4–8) and keep the dialog box.

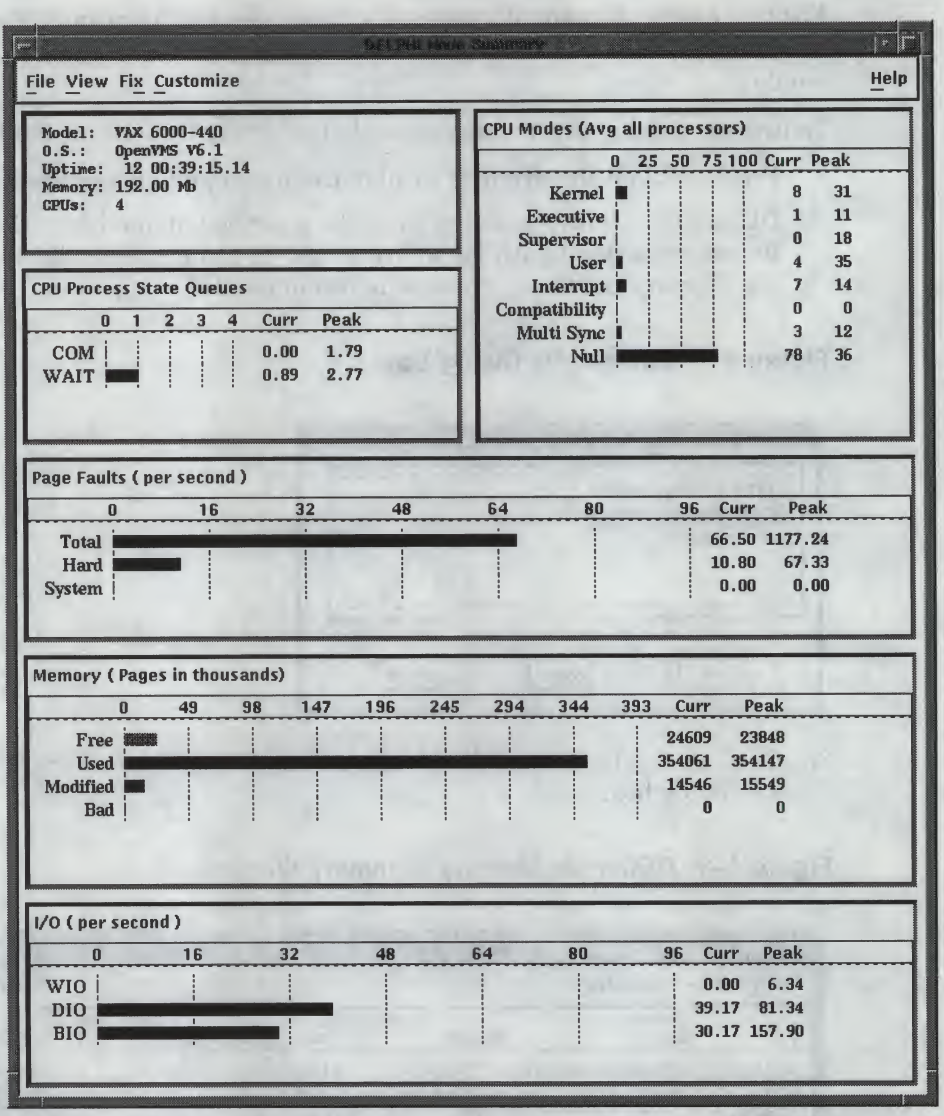
Figure 4–8 DECamds Memory Summary Window



3. To determine which process consumes the most memory and is not page faulting, you can sort and examine the data in the Memory Summary window. In this example, the process DECW$SERVER_0 is consuming the most memory and not page faulting.

4. Select the Node Summary window from the Low Memory dialog box and click on Apply to display the window. DECamds displays a window similar to Figure 4–9.

Figure 4–9 DECamds Node Summary Window



The Node Summary window in Figure 4–9 confirms there is little free memory available. (The Node Summary window also can show other activity relevant to diagnosing the problem such as high page faults.)

5. Purge the working sets. You choose which process's working sets are to be purged by performing the following steps:

   a. From the Memory Summary window, select any process, press MB3 on the count field, and choose Fix from the menu.

   b. Click on OK or Apply in the Fix dialog box.

5
Customizing DECamds

This chapter describes how to organize data collection, analysis, and display by filtering, sorting, and customizing DECamds. It also describes how some of these tasks can optimize the performance of DECamds.

5.1 Customizing DECamds Defaults

To set DECamds application values such as bar graph colors and automatic collection options, choose DECamds Customizations from the Customize menu of the Event Log and System Overview windows. DECamds displays the DECamds Applications Customizations dialog box, Figure 5–1.

Figure 5–1 DECamds Application Customizations Dialog Box

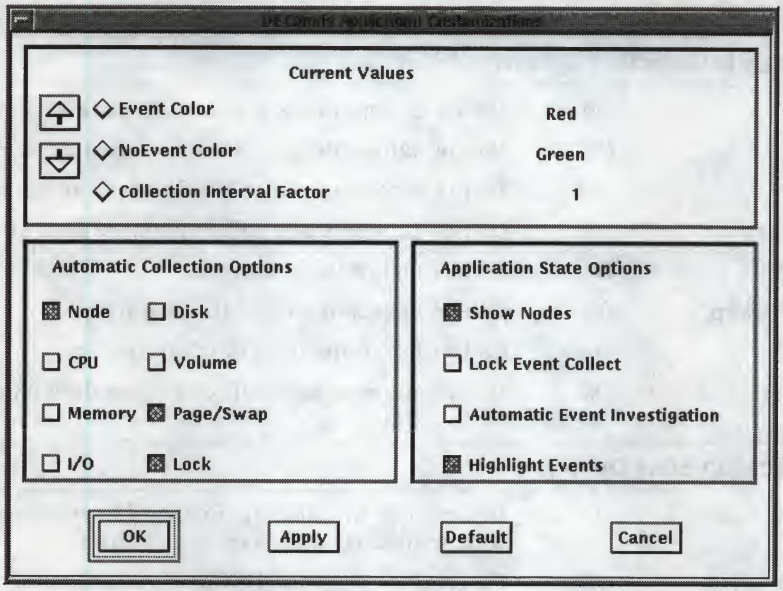


Table 5–1 lists the items you can customize.

To save your changes from use to use, choose Save DECamds Customizations from the Customize Menu of the Event Log or System Overview window. The changes are stored in the AMDS$APPLIC_CUSTOMIZE.DAT file.

Note

Subsequent installations of DECamds will not overwrite existing customization files. The installation procedure will check for the existence of each of the customization files. If found, the procedure will provide the new file with the .TEMPLATE file extension. The installer must check the new .TEMPLATE files for new features implemented in future releases; any changes will be stated in the online release notes in the following location:

```
SYS$HELP:AMDS062.RELEASE_NOTES
```

Table 5–1 DECamds Application Defaults

Field	Default	Function
Current Values		
Event Color	Red	Specifies the bar graph color used for signaled events.
No Event Color	Green	Specifies the bar graph color used for non-signaled events.
Collection Interval Factor	1	This value is multiplied by a window's collection interval definition. Used to force windows to have longer time spans between data collection. Increasing this number decreases the use of the Data Analyzer's CPU and LAN.
Automatic Collection Options		
CPU	Off	Do not automatically collect CPU data at startup.
Disk	Off	Do not automatically collect disk data at startup.
I/O	Off	Do not automatically collect I/O data at startup.
Memory	Off	Do not automatically collect memory data at startup.
Node	On	Collect node data at startup.
Page/Swap	On	Collect page and swap data at startup.
Lock	On	Collect lock contention data at startup.
Volume	Off	Do not automatically collect volume data on startup.
Application State Options		
Show Nodes	On	Determines whether the System Overview window starts up with individual node names displayed.
Lock Event Collect	Off	Determines whether DECamds automatically collects additional data about all the processes waiting for a locked resource.
Automatic Event Investigation	Off	Determines whether additional data is collected when DECamds detects an event.
Highlight Events	On	Determines whether event-related data is highlighted.

5.1.1 Setting Default Data Collection

By default, DECamds collects the following categories of data when started: Node Summary, Page/Swap File Summary, and Lock Contention Summary.

You can change the default amount of data collected when DECamds starts by choosing Customizations from the Customize menu in the Event Log or System Overview windows. The DECamds Application Customizations dialog box appears and you can click on the Automatic Collection Options buttons to select or disable the categories you want. To save the settings for the next time you run DECamds, choose Save DECamds Customizations from the Customize menu.

5.1.2 Setting Automatic Event Investigation

Automatic Event Investigation enhances the speed with which you can pursue a specified event. When this option is enabled, DECamds automatically collects follow-up data on the event. When this option is disabled, you must initiate follow-up data collection when an event occurs.

To enable automatic event investigation, choose Enable Automatic Event Investigation from the Control menu of the System Overview or Event Log windows. To disable it, choose the Disable Automatic Event Investigation menu item.

You also can set Automatic Event Investigation by choosing Customizations from the Customize menu, then click on the Automatic Event Investigation button in the resulting DECamds Application Customizations dialog box. To save the settings for the next time you run DECamds, choose Save DECamds Customization from the Customize menu.

Note

Enabling this option can significantly increase CPU, memory, and LAN traffic load. By default, DECamds does not automatically investigate events that may require attention.

Also, Automatic investigation only applies to events that are detected after you enable the option. It does not apply to lock-related events. To control automatic investigation for lock-related events, use the DECamds Customizations dialog box

5.1.3 Setting Automatic Lock Investigation

Automatic Lock Investigation is a feature where the Data Analyzer will automatically investigate any signalled Lock Contention events. Setting this option allows you to determine more quickly the blocking lock in a resource contention situation.

Note

This option uses more DECamds memory, CPU, and LAN bandwidth to sometimes investigate locks that are very transient.

To enable automatic investigation of locks, click on the Lock Event Collect button in the DECamds Application Customization dialog box.

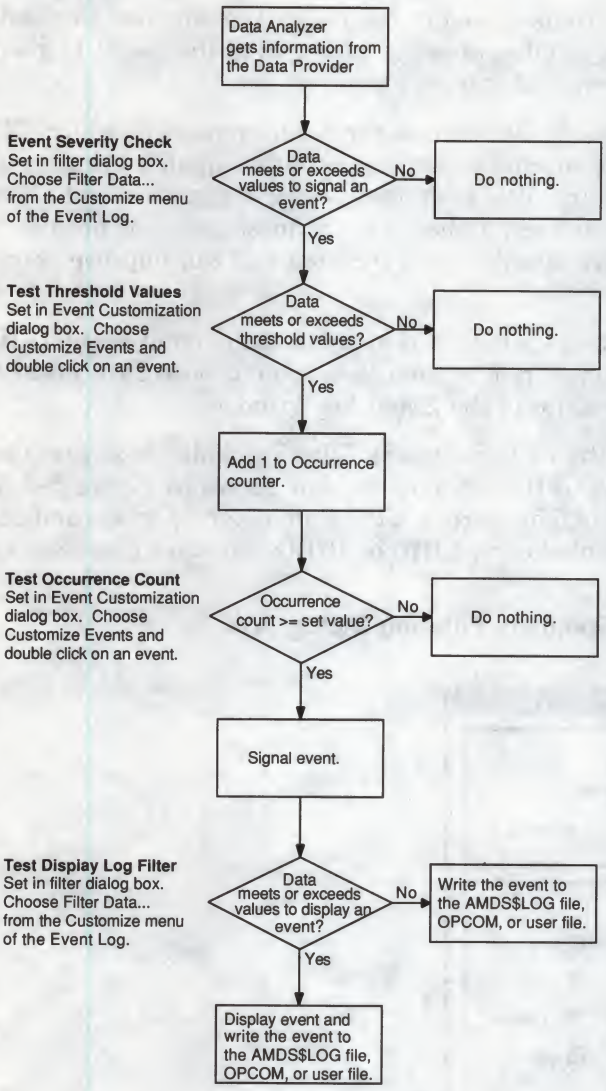
5.2 Filtering Data

DECamds can collect and display every event regardless of how important or unimportant an event is to you. However, you will want to narrow the focus so that the events that you want to see are displayed. There are two mechnisms for determining which events qualify for your attention.

- Filtering all events on a global severity basis. For example, you may not want to see any event that has less than a 40 severity value.
- Defining specific event criteria. For example, you can refine the global filtering by also defining that DSKRWT event (high disk device Rwait count) must pass your specifications before being considered an event worth displaying or logging.

Figure 5–2 shows the process an event must pass through for it to qualify as important enough to be logged or displayed for your attention.

Figure 5–2 Event Qualification



5.2.1 Filtering Events

To determine the events that can be displayed, perform the following steps:

1. Choose Filter Data... from the Customize menu. A filter dialog box appears. You can filter data in the following data windows:

   CPU Summary
   Lock Contention Summary
   Memory Summary
   Process I/O Summary
   Disk Status Summary
   Disk Volume Summary
   Page/Swap File Summary

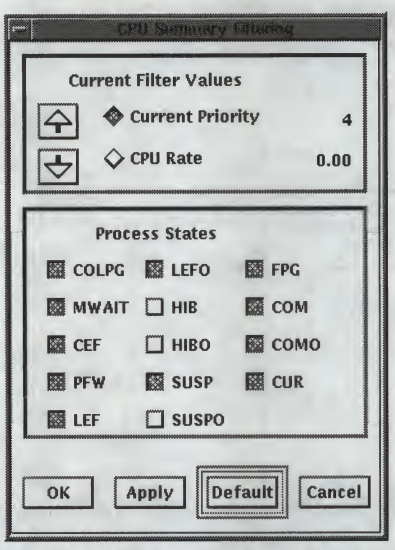
   The filter dialog box has the following modifable options:

- Severity — Determines the severity value at which DECamds writes an event in the Event Log window. Only events that meet or exceed this value are displayed in the Event Log window. Increasing this value reduces the number of event messages in the Event Log window and can improve perceived response time.
- Event Signal — Determines the severity value at which DECamds signals an event for attention. Only events that qualify are passed on to be checked by any filters you may set for a specific event. Increasing this value reduces the number of event messages that need to be tested to see if further attention is warranted and can improve perceived response time.

2. Modify the settings, which will apply to the current session. If you want to save these settings from session to session, choose Save Filter Changes from the Customize menu in the Event Log Window.

Figure 5–3 shows the CPU Summary Filtering dialog box. For a process to be displayed using the CPU Summary window shown in Figure 5–3, it must have a Current Priority of 5 or more, a CPU Rate of 2.5 or more, and be in any of the process states indicated except HIB or HIBO. No other processes are displayed.

Figure 5–3 CPU Summary Filtering Dialog Box



If the Enable Highlighting option is on, any process that signals an event is included in the display, regardless of whether it meets the filter criteria.

To change the value of a filter, click the filter button on, then click the up or down arrow. Click on OK or Apply for the filter to take effect. To return to system default values, click on Default.

Changing a Filter Category

Some data windows also allow you to filter data by category. For example, in the CPU Summary window, you also can filter by the Process State category to display only processes in certain states. Category buttons that are selected display the associated information.

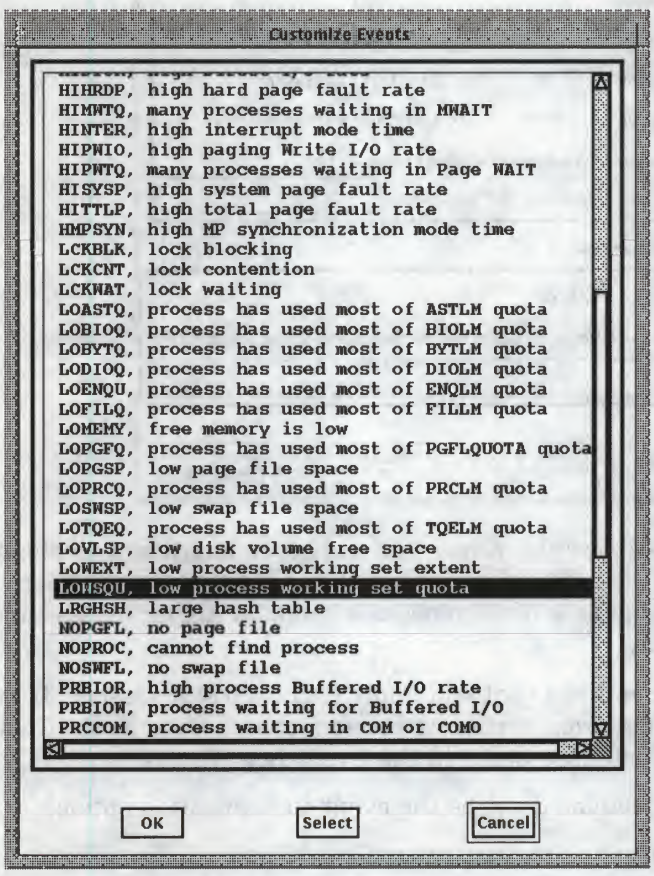
In the CPU Summary window, to display only inactive processes, select the HIB and HIBO buttons under Process States, and deselect all other process states. When you click on OK or Apply, only inactive processes appear in the CPU Summary window.

5.2.2 Customizing Events

You can define criteria by which specific events are qualified for your attention. For example, you can refine the global filtering by also defining that DSKRWT event (high disk device Rwait count) must pass your specifications before being considered an event worth displaying or logging. To define specific event criteria, perform the following steps:

1. Choose Customize Events from the Customize menu in the Event Log Window. Figure 5–4 shows the Customize Events dialog box that appears.

Figure 5–4 Customize Events Dialog Box



2. Double click on an event that you want to customize. A dialog box appears with the event you selected. The dialog box also contains an explanation of what might cause this event to occur. Figure 5–5 shows the LOWSQU Event Customization Window.

Figure 5–5 LOWSQU Event Customization Window

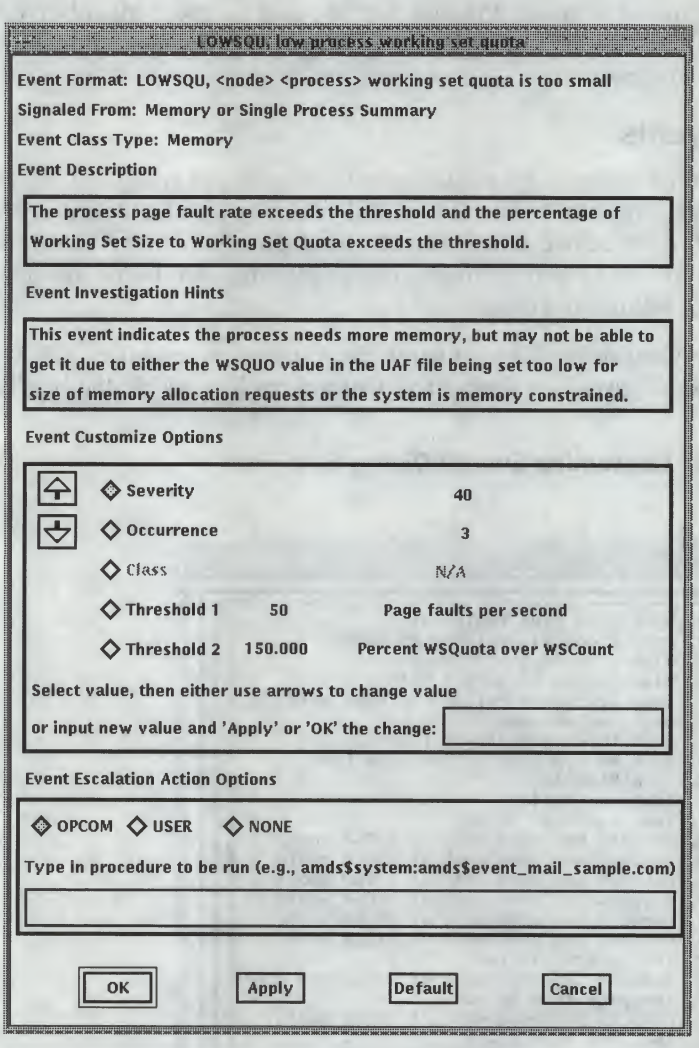


Figure 5–5 shows the values you can set in any Event Customization window. To change the value of an option, click on an option and then use the arrow buttons to increase or decrease the value. A higher number indicates a more severe event.

3. Modify the settings that will apply to the current session. If you want to save these settings from session to session, choose Save Event Customizations from the Customize menu in the Event Log Window.

The following sections describe the event customization options.

Severity Option

Severity is the relative importance of an event. Events with a high severity must also exceed threshold settings before an event can be signaled for display or logging.

Occurrence Option

Each DECamds event is assigned an **occurrence** value, that is, the number of consecutive data samples that must exceed the event threshold before the event is signaled. By default, events have low occurrence values. However, you might find that a certain event only indicates a problem when it occurs repeatedly for an extended period. You can change the occurrence value assigned to that event so that DECamds signals it only when necessary.

For example, suppose page fault spikes are common in your environment, and DECamds frequently signals intermittent *HITTLP, total page fault rate is high* events. You could change the event's occurrence value to 3, so that the total page fault rate must exceed the threshold for three consecutive collection intervals before being signaled to the Event Log.

To avoid displaying insignificant events, you can customize an event so that DECamds signals it only when it continuously occurs.

Automatic Event Investigation (see Section 5.1.2) uses the Occurrence value to determine when to further investigate an event. When enabled, the automatic event investigation is activated when the Occurrence count is three times the Occurrence setting value.

Class Option

You can customize certain events so that the event threshold varies depending on the class of computer system the event occurs on. This feature is particularly useful in environments with many different types and sizes of computers.

By default, DECamds uses only one default threshold for each event, regardless of the type of computer the event occurs on. However, for certain events (in particular, CPU, I/O, and memory usage events) the level at which resource use becomes a problem depends on the size and type of computer. For example, a page fault rate of 100 may be important on a VAXstation 2000 system but not on a VAX 7000 system.

DECamds provides three additional predefined classes for CPU, I/O, and Memory-related events. You can specify threshold values for each class in addition to the default threshold for an event. To specify an additional event threshold for each class, edit the file AMDS$THRESHOLD_DEFS.DAT located in the AMDS$CONFIG directory. An example is provided at the end of this section.

Table 5–2 defines CPU, I/O, and Memory classes.

Table 5–2 CPU, I/O, and Memory Class Definitions

Class[1]	Description
	CPU Classes
Class 1	All VAXft systems, VAXstation/VAXserver 4000, MicroVAX 4000
Class 2	Higher VUP workstations: VAXstation/VAXserver 3100-M76, MicroVAX 3100-M76, MicroVAX 3100-8*, VAXstation 3100-9*, MicroVAX 3100-9*, VAXstation 4000-9*
Class 3	VAX/VAXserver 6000, 7000, 9000, 10000
Class 4	All Alpha systems

[1]If no class is defined, DECamds uses the default threshold value.

(continued on next page)

Table 5–2 (Cont.) CPU, I/O, and Memory Class Definitions

Class[1]	Description
	I/O Classes
Class 1	All VAX systems, All VAXft systems, VAXstation/VAXserver 4000, MicroVAX 4000
Class 2	Higher VUP workstations: VAXstation/VAXserver 3100-M76, MicroVAX 3100-M76, MicroVAX 3100-8*, VAXstation 3100-9*, MicroVAX 3100-9*, VAXstation 4000-9*
Class 3	VAX/VAXserver 6000, 7000, 9000, 10000
Class 4	All Alpha systems
	Memory Classes
Class 1	Systems with less than or equal to 24 MB of memory
Class 2	Systems with more than 24 MB and less than or equal to 64 MB of memory
Class 3	Systems with more than 64 MB of memory
Class 4	All Alpha systems

[1]If no class is defined, DECamds uses the default threshold value.

You can specify class-based thresholds only for the following events:

- CPU-related events:
  - HINTER, *node* interrupt mode time is high
  - HICOMQ, *node* many processes waiting for CPU
  - HMPSYN, *node* MP synchronization mode time is high
  - HIPWTQ, *node* many processes waiting in COLPG, PFW, or FPG
  - HIMWTQ, *node* many processes waiting in MWAIT
- I/O-related events:
  - HIBIOR, *node* buffered I/O rate is high
  - HIDIOR, *node* direct I/O rate is high
  - HIPWIO, *node* paging write I/O rate is high
- Memory-related events:
  - LOMEMY, *node* free memory is low
  - HIHRDP, *node* hard page fault rate is high
  - HISYSP, *node* high system page fault rate
  - HITTLP, *node* total page fault rate is high
  - RESPRS, *node* resource hash table sparse
  - RESDNS, *node* resource hash table dense

As an example of setting a class-based threshold, the *HITTLP, total page fault rate is high* event is a memory-related event, so the thresholds are based on the memory class definitions shown in Table 5–2. The default threshold for this event is 20 page faults per second. A page fault rate of 20 may be important on a VAXstation 2000 system but it is not important on a VAX 7000 system. To account for this, you can specify the following additional thresholds for the *HITTLP, total page fault rate is high* event:

Class	Threshold	Description
1 (systems with less than or equal to 64 MB of memory)	20	Event is triggered at the default threshold of 20 page faults per second.
2 (systems with 24 MB to 64 MB of memory)	40	Event is triggered at 40 page faults per second.
3 (systems with more than 64 MB of memory)	100	Event is triggered at 100 page faults per second.
4	100 (Alpha systems)	Event is triggered at 100 page faults per second

Threshold Options

Threshold values are compared to an event's description to determine whether an event meets the criteria for display or log. Threshold values are used in conjunction with the Occurrence and severity values. Increasing event threshold values can reduce CPU use and improve perceived response time because more instances must occur for the threshold to be crossed, so fewer thresholds are crossed and fewer events are triggered.

Note

Setting a threshold too high could mask a serious problem.

You can read a description of an event by choosing Customize Events from the Customize menu in the Event Log window, then double clicking on the event. The event customization dialog box displays an Event Description field.

Most events are checked against only one threshold; however, some have dual thresholds, where the event is triggered if either one is true. For example, for the *LOVLSP, node disk volume free space is low* event DECamds checks both of the following thresholds:

- Number of blocks remaining (*LowDiskFreeSpace.BlkRem*)
- Percentage of total blocks remaining (*LowDiskFreeSpace.Percent*)

Note

Events with both high severity and threshold values are signaled to the Operator Communication Manager (OPCOM). For more information about signaling events to OPCOM, see Section 2.3.3.

5.3 Sorting Data

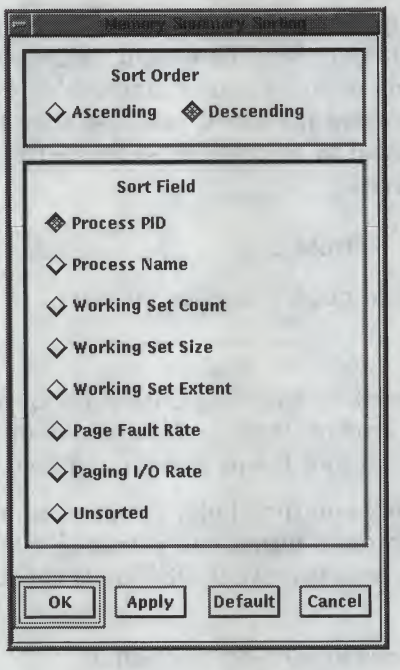
Choose Sort Data... from the Customize menu to change the order of the information displayed in a window. A dialog box appears in which you can specify sort criteria. All sort criteria must be met for a process to be displayed.

You can sort data in the following windows:

CPU Summary
Disk Status Summary
Disk Volume Summary
Event Log
Lock Contention Summary
Memory Summary
Page/Swap File Summary
Process I/O Summary

Figure 5–6 shows a sample Memory Summary Sorting dialog box.

Figure 5–6 Memory Summary Sorting Dialog Box



Sorting is based on two variables: the sort field and the sort order. You can choose only one sort criterion for each variable. For example, to sort Memory Summary data to list the processes with the highest page fault rates first, perform the following steps:

1. Choose Sort Data... from the Customize menu on the Memory Summary window. The Memory Summary Sorting dialog box appears; current sort field settings are displayed. (By default, DECamds sorts Memory Summary data on the Working Set Count field in descending order.)
2. Change sort settings by choosing Page Fault Rate and Ascending order.
3. Click on OK or Apply.

5.4 Setting Collection Intervals

A **collection interval** is the time the Data Analyzer waits before requesting more information from Data Provider nodes. Changing the collection interval helps you control DECamds' performance and consumption of system resources.

The frequency of polling remote nodes for data (collection intervals) can affect perceived response time. You want to find a balance between collecting data often enough to detect potential resource availability problems before a node or cluster experiences a severe problem, and seldom enough to optimize perceived response time. Increasing the collection interval factor decreases CPU consumption and LAN load, but response time might appear slower.

Collection intervals do not affect memory use.

To change a collection interval, choose Collection Interval from the Customize Menu. Figure 5–7 shows a sample Memory Summary Collection Interval dialog box.

Figure 5–7 Memory Summary Collection Interval Dialog Box

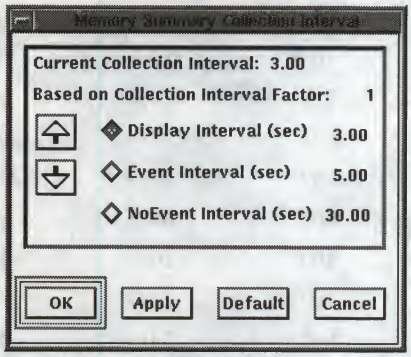


Table 5–3 describes the fields on the Memory Summary Collection Interval dialog box.

Table 5–3 Memory Summary Collection Interval Fields

Current Collection Interval	Displays the number of seconds between requests for data. You can change the value for all collection intervals for all windows by choosing DECamds Customizations from the Customize menu of the Event Log or System Overview window. The DECamds Application Customizations dialog box appears and you can increase or decrease the collection interval factor.
Based on Collection Interval Factor	Displays the number with which the collection interval is multiplied.
Display Interval (sec)	Displays the collection interval for displaying data in a window. You can change the interval by clicking on the up or down arrows in the dialog box.
Event Interval (sec)	Displays the collection interval used when events are found. This value is used by default when you start background collection. You can change the interval by clicking on the up or down arrows in the dialog box.

(continued on next page)

Table 5–3 (Cont.) Memory Summary Collection Interval Fields

NoEvent Interval (sec)	Displays the collection interval when no events are found. You can change the interval by clicking on the up or down arrows in the dialog box.

To apply the changes, click on OK or Apply. To change back to DECamds default values for the window, click on Default. To exit without making any changes, click on Cancel.

Table 5–4 lists the default window interval values (in seconds) provided with DECamds for each window type.

Table 5–4 Default Window Collection Intervals

Window	Display[1]	Event[1]	No Event[1]
CPU Modes Summary	5.0	5.0	5.0
CPU Summary	5.0	10.0	30.0
Disk Status Summary	30.0	15.0	60.0
Disk Volume Summary	15.0	15.0	120.0
Lock Contention	10.0	20.0	60.0
Memory Summary	5.0	10.0	30.0
Node Summary	5.0	5.0	10.0
Page/Swap File Summary	30.0	30.0	2400.0
Process Identification Manager[2]	60.0	60.0	240.0
Process I/O Summary	10.0	10.0	30.0
Single Lock Summary	10.0	10.0	20.0
Single Process Summary	5.0	5.0	20.0

[1]All times are in seconds and cannot go below .5 seconds.

[2]Process Identification Manager supports the CPU, Memory, Process I/O, and Single Lock Summary window sampling.

5.5 Optimizing Performance with System Settings

DECamds is a compute-intensive and LAN traffic-intensive application. At times, routine data collection and display activities can cause a delay in perceived response time.

This section explains how to optimize perceived response time based on actual measurements of CPU utilization rates (throughput). Performance improvements can be made in the following areas:

- DECamds software
- System settings
- Hardware configuration

Site configurations vary widely and no rules cover all situations but the information in this section can help you make informed choices about improving your system performance. The following factors affect perceived response time:

- Load on monitored nodes including applications and peripherals (especially number of disks)
- Number of monitored nodes and users
- Size of operating system tables and lists on monitored nodes (process and lock)
- Version of operating system running on monitored nodes
- LAN traffic, cluster communications, nodes booting, and network-based applications and tools

When DECamds starts, it polls the local area network (LAN) to locate all nodes running the DECamds Data Provider, creates a communications link, and collects data from each Data Provider node on the LAN. (See Section 1.3 for more information about establishing a communications link between nodes.)

The initial polling process creates a short-term high load of CPU and LAN activity. After establishing a communications link with other nodes, DECamds reduces polling frequency, thereby reducing the CPU and LAN load.

Note

Each request to collect a new category of data increases memory and LAN requirements. Memory requirements vary with the number of categories collected and the number of nodes being polled.

Polling frequency does not affect memory because polling only changes how frequently existing data is replaced with updated data.

Using DECamds to observe nodes or to take corrective action can affect performance. The following sections describe how you can improve perceived response time.

The following sections also describe system settings that you can make to improve DECamds's ability to handle the data collection demands.

5.5.1 Setting Process Quotas

To improve the performance of DECamds, you may need to change process quotas, using the following steps:

1. Increase the process quotas assigned to the process initiating DECamds in the system's user authorization file (UAF). The quotas used extensively by DECamds are ASTLM, TQELM, BIOLM, BYTLM, and WSEXTENT. The values in the installation guide are suggestions for a 50-node cluster. The following process quotas are recommended:

Quota	Recommended Value[1]
ASTLM	4 times the node count
TQELM	4 times the node count
BIOLM	2 times the node count
WSEXTENT	350 times the node count
BYTLM	1500 times the node count

[1] *node count* is the number of nodes a Data Analyzer monitors simultaneously.

2. Log out, log back in, and restart DECamds.

5.5.2 Setting LAN Load

The maximum size for data packets is 1500 bytes. When the amount of data is greater than 1500 bytes, DECamds must send multiple requests to complete the data collection request.

Table 5–5 shows the LAN load for various levels of collection intervals and data collection. You can modify a data collection window's collection intervals (as explained in Section 5.4) or reduce the scope of data collection (as explained in Section 5.1.1) to reduce LAN activity.

Table 5–5 LAN Load

Data	Outgoing Packet Size (in bytes) on Alpha	Outgoing Packet Size (in bytes) on VAX	Return Packet Size (in bytes)
Configuration data	129	285	88
CPU Modes	201	129	48 + (64* no. of processors)
CPU Summary	178	171	16 per active process
Disk Status Summary	473	473	56 per disk
Fix	24	24	12
Hello message	N/A	N/A	32
Lock Contention	240	240	76 per resource
Memory Summary	275	275	36 per active process
Node Summary	319	241	48 + (64 * no. of processors)
Page/Swap File	208	208	46 per page/swap file
Process I/O Summary	236	229	32 per active process
Single Lock (Waiting)	272	272	32 per waiter
Single Process Summary	491	471	00
Volume Summary	430	430	28 per disk

5.5.3 Setting Data Link Read Operations

Increase read operations to the data link by changing the logical name AMDS$COMM_READS in the AMDS$CONFIG:AMDS$LOGICALS.COM command procedure. The AMDS$COMM_READS logical name controls the number of requests for data (read operations) queued to the data link.

If you increase data collection, increase the number of requests that can be queued. Digital recommends two for each node being monitored. Each read operation queued requires 1500 bytes of BYTLM quota.

5.5.4 Setting the Communications Buffer

Increase the communications buffer by changing the logical name AMDS$COMM_BUFFER_SIZE in the AMDS$CONFIG:AMDS$LOGICALS.COM command procedure. The buffer controls the size of the global section used for communication between the collector node and the communications process.

When DECamds cannot keep up, you will receive the following warning message:

```
AMDS$_COMMBUFOVF---communications buffer overflow.  Increase the buffer by 25 percent.
```

In addition to increasing the value of the AMDS$COMM_BUFFER_SIZE logical name, set the system parameter GBLPAGFIL on the collector node to cover the increase. This adds to the amount of data collection that DECamds can perform.

The value of the GBLPAGFIL system parameter must always be higher than the number of FREE_GBLPAGES. To determine the value of FREE_GBLPAGES:

```
$ A = F$GETSYI("FREE_GBLPAGES")
$ SHO SYM A
```

The value of A must conform to the formula:

```
2 * ( (buffer_size / 512) + 512)
```

5.5.5 Setting Window Customizations

The Sort, Filter, and collection interval settings at the data window level affect performance. Follow these guidelines to balance customization with performance:

- Sorting — Use unsorted windows to improve performance. Sorting requires extra computations.
- Changing window filters — Filter out data to improve CPU performance. Reducing the collection criteria increases performance.

  To filter data, choose Filter from the Customize menu, then select filter options from the Filter Data dialog box. To save filter changes, choose Save Filter Changes from the Customize menu before stopping data collection.
- Changing window collection intervals — Increase collection interval values to improve performance (see Section 5.4). To save collection interval changes, choose Save Collection Interval Changes from the Customize menu before stopping data collection.

5.6 Optimizing Performance with Hardware

Table 5–6 provides an approximate guideline for the number of nodes you can monitor when running DECamds on certain computer types:

Table 5–6 Monitoring Nodes

Monitoring Computer Type	Number of Nodes Monitored	
	Observation Only	Observe and Fix
VAXstation 3100	0-30	0-20
VAXstation 4000 Model 60	20-60	20-50
VAX 6000[1]	75-150	65-130
VAX 4000 Model 90	75-150	65-130
DEC 3000 Model 400	0-50	0-50
DEC 4000 Model 620	0-70	0-70

[1]With DECwindows display directed to a workstation

(continued on next page)

Table 5–6 (Cont.) Monitoring Nodes

Monitoring Computer Type	Number of Nodes Monitored	
	Observation Only	Observe and Fix
DEC 7000 Model 720	Any number	Any number

Follow these suggestions when choosing and configuring a console:

- Use fast hardware.

  Because DECamds is compute and memory-intensive, which is compounded by its real-time, DECwindows based display medium, faster CPUs will improve throughput and perceived response time.

- Use multiprocessors.

  DECamds runs two processes: one handling calculations and display; one handling communications between the monitoring node and the remote nodes. A multiprocessor reduces the DECwindows server process competition for CPU time. On single processor systems, the processes must compete.

- Run the monitoring portion of DECamds on a standalone system.

  This avoids the possibility of being unable to use DECamds if a cluster is experiencing system resource problems.

A

Installing DECamds

This appendix describes how to install DECamds software on OpenVMS Alpha and VAX systems.

A.1 Preparing to Install DECamds

This section describes the preparations and requirements necessary for installing DECamds.

DECamds provides online release notes. Digital strongly recommends that you read the release notes before proceeding with the installation. You can print the release notes from the following location:

```
SYS$HELP:AMDS062.RELEASE_NOTES
```

For information about how to access the online release notes, see Section A.5.

A.2 Kit Location and Installation Information

The DECamds kit has two components:

- The Data Analyzer is installed on any system where you want to analyze, detect and display events, and initiate corrective action.
- The Data Provider is installed on any node from which you want to gather system data and permit fixes.

Alpha

For OpenVMS Alpha systems, use the POLYCENTER Software Installation Utility to install the DEC-AXPVMS-AMDS-V0602–1.PCSI kit (see Section A.4) on the software distribution compact disk.♦

VAX

For OpenVMS VAX systems, install the AMDS062.A and AMDS062.B save sets using the VMSINSTAL procedure. The kits are located in the [AMDS062] directory on the compact disk.

For information on how to install and use DECamds on OpenVMS VAX systems, see Section A.5. You can find the DECamds kits, named AMDS062.A and AMDS062.B, on the following media:

Magnetic tape	Third volume
TK50 tape	Second volume
Compact disk	[AMDS062] directory♦

A.3 Installation Requirements

This section provides a checklist of hardward and software requirements for DECamds. A typical installation takes approximately 5 to 10 minutes per node, depending on the type of media and your system configuration.

- **Hardware Requirements**
  - A workstation monitor. For any hardware configuration without a DECwindows Motif display device, use the DECwindows server to direct the display to a workstation or an X terminal.
  - 16 MB of memory for VAX systems and 32 MB for Alpha systems, for the Data Analyzer portion of DECamds.

  You should use a more powerful system as the number of nodes and the amount of collected data rises. Table A–1 shows general guidelines for the default Data Analyzer node. Note that the following table does not preclude DECamds from running on a less powerful system than listed for the number of nodes being monitored.

Table A–1 Recommended System Requirements

Number of Monitored Nodes	Recommended Alpha Hardware	Recommended VAX Hardware
1-30	DEC 3000 Model 400, 32 MB	VAXstation 3100, 16 MB
20-50	DEC 3000 Model 400, 64 MB	VAXstation 4000 Model 60
40-90	DEC 3000 Model 500	VAXstation 4000 Model 90
91 or more	DEC 4000 Model 620	VAX 6000-420

- **Operating System Version**

  At least one of the following:

  OpenVMS Version 5.5 or higher
  OpenVMS VAX V6.0 or higher
  OpenVMS AXP V6.1
  OpenVMS Alpha V6.2 or higher

- **Display Software**

  DECwindows Motif for OpenVMS Version 1.1 or higher installed on the Data Analyzer node system.

- **Privileges**

Operation	Privileges Needed
Monitor only (read-only access)	OPER
Implement fixes (write access)	OPER, CMKRNL
Stop, start, reload, or restart the Data Provider node. Includes changing security or group name.	OPER, CMKRNL, LOG_IO, SYSNAM, SYSPRV

Note

For OpenVMS VAX Version 6.0 and later or OpenVMS AXP Version 6.1 and later, if the Data Provider is running on the same node as the Data Analyzer node, you must also have either SYSPRV privilege or ACL access to the RMA0: device.

- **License Registration**

  VAXcluster, VMScluster, or VMScluster-Client license. For information on registering a license, see the *OpenVMS License Management Utility Manual*.

- **Disk Space**

  Alpha

  – 4000 blocks on Alpha.♦

  VAX

  – 3500 blocks on VAX. Note that VMSINSTAL turns off BYPASS privilege at the start of the installation.♦

  To determine the number of free disk blocks on the current system disk, enter the following command at the DCL prompt:

```
$ SHOW DEVICE SYS$SYSDEVICE
```

- **System Parameter Settings**

  Same as needed for operating system installation. The Installation Verification Procedure (IVP) requires additional space as follows:

  – GLBPAGFIL 1200
  – WSMAX 16384

  You can modify WSMAX and GLBPAGFIL using the System Management utility (SYSMAN). See the *OpenVMS System Manager's Manual* for more information.

- **Process Account Quotas** (minimum)

  – ASTLM[1] 150
  – BIOLM[1] 51
  – BYTLM[1] 75000
  – FILLM 20
  – PRCLM 3
  – PGFLQUO 25600
  – TQELM[1] 100
  – WSEXTENT 16384

[1]The AMDS$COMM_READS logical determines the default value. If you are reinstalling DECamds, or have changed AMDS$COMM_READS, then the following formulas are used to determine the default value:

```
ASTLM >= (AMDS$COMM_READS*3)
BIOLM >= (AMDS$COMM_READS+1)
BYTLM >= (AMDS$COMM_READS*1500)
TQELM >= (AMDS$COMM_READS*2)
```

User account quotas are stored in the file SYSUAF.DAT. Use the OpenVMS Authorize utility (AUTHORIZE) to verify and change user account quotas. For more information on modifying account quotas, see the description of the Authorize utility in the OpenVMS system management documentation.

- **VMSINSTAL Requirements (VAX only)**

VAX

  - Log in to a privileged account.
  - Ensure that the default device and directory is set to the SYS$UPDATE logical name.
  - Ensure adequate quotas for installation.
  - Ensure that users are not logged in to the system.

  If VMSINSTAL detects any problems during the installation, it notifies you and asks if you want to continue the installation. In some instances, you can enter YES to continue. To stop the installation process and correct the situation, enter NO and press the Return key; then correct the problem and restart the installation.♦

Note

Digital recommends that you perform a system disk backup before installing any software. Use the backup procedures that are established at your site. For details about performing a system disk backup, see the OpenVMS Backup utility documentation.

A.4 OpenVMS Alpha System Installation

Alpha

This section describes the installation procedure on an OpenVMS Alpha system.

- **Starting the Installation**

  Depending on the interface to your system, use one of these procedures to start the installation:

Interface	Command
Motif	From a DECterm, enter the PRODUCT command at the DCL prompt ($): `$ PRODUCT`
DCL	Enter the PRODUCT command, the name of the task to be performed, and the name of one or more products. For example, to install DECamds Version 6.1, enter the following command: `$ PRODUCT INSTALL AMDS/SOURCE=distribution-media/HELP`

For a description of the features you can request with the PRODUCT INSTALL command when starting an installation such as running the IVP, purging files, and configuring the installation, see the *POLYCENTER Software Installation Utility User's Guide* or Motif help.

As an installation procedure progresses, the system displays a percentage message to indicate how much of the installation is done. For example:

```
Percent Done: 15%
...30%
...46%
...62%
...76%
...92%
%PCSI-I-SUCCESS, operation completed successfully
```

If you started the installation using the /LOG qualifier, the system displays details of the installation.

- **Stopping and Restarting the Installation**

  Depending on the interface to your system, use one of these procedures to stop and restart the installation:

Interface	Action
DCL	To stop the procedure at any time: press Ctrl/Y and then enter the PRODUCT REMOVE command to reverse any changes to the system that occurred during the partial installation. This deletes all files created up to that point and causes the installation procedure to exit.
Motif	To close the POLYCENTER Software Installation utility window, choose Exit from the File menu.

- **Recovering from Errors**

  If the installation procedure fails for any reason, the following message is displayed:

```
%POLYCENTER Software Installation utility
%INSTAL-E-INSFAIL, The installation of DECamds 6.2 has failed.
```

  An error during the installation can occur if one or more of the following conditions exist:

  - The operating system version is incorrect.
  - The prerequisite software version is incorrect.
  - Quotas necessary for successful installation are inadequate.
  - Process quotas required by the POLYCENTER Software Installation utility are inadequate.
  - The OpenVMS Help library is currently in use.

  If you receive any error message beginning with `%PCSI-E-INSTAL`, refer to the DCL HELP/MESSAGE utility for POLYCENTER Software Installation information and a possible solution to the problem.

  If the installation fails, you must restart the installation procedure. If the installation fails due to an IVP failure, contact a Digital field service representative.

Sample Installation on an OpenVMS Alpha System

Example A–1 shows a sample installation on an OpenVMS Alpha system. This sample was run on a system that had no previous version of DECamds installed. Depending on which layered products you have on your system, you might see additional messages and questions when you perform your installation.

Example A–1 Sample OpenVMS Alpha Installation

```
$ PRODUCT INSTALL AMDS/SOURCE=DAD15:[AMDS062.KIT]

The following product has been selected:
DEC AXPVMS AMDS V6.2     [Available]

Do you want to continue? [YES]

*** DEC AXPVMS AMDS V6.2: DECamds (Availability Manager for Distributed Systems) V6.2

    COPYRIGHT © (c) 1994, 1995 -- All rights reserved

    Digital Equipment Corporation

    License and Product Authorization Key (PAK) Information

    Do you want all the default values for this product? [YES]

    DECamds Startup File

    DECamds Message Help

    DECamds Logicals Customization File

    DECamds Data Provider Security Access File

    DECamds Data Provider Installation Verification Procedure

    DECamds Data Analyzer Security Access File

    DECamds Data Analyzer Installation Verification Procedure (IVP)

    IVP may fail due to the following PQL values being too low:

    PQL_MASTLM, PQL_MBIOLM, PQL_MTQELM, or PQL_MBYTLM

    See the file AMDS$SYSTEM:AMDS$PCSI_IVP_OUTPUT.LOG for help on failure.

    Do you want to view the values? [NO]

%PCSIUI-I-DONEASK, execution phase starting
The following product will be installed:
DEC AXPVMS AMDS V6.2
%PCSI-I-VOLINFO, estimated space information for volume DISK$ALPHAVMS061
-PCSI-I-VOLSPC, 0 required; 89505 available; 89505 net

Portion Done: 0%...30%...40%...50%...60%...70%...80%...90%...100%
The following product has been installed:
DEC AXPVMS AMDS V6.2
%PCSI-I-EXETSTSTART, start of test procedure
%PCSI-I-EXETSTOK, end of test procedure; completed with no errors

*** DEC AXPVMS AMDS V6.2: DECamds (Availability Manager for Distributed Systems) V6.2

    This product requires the following SYSGEN parameters:
        GBLPAGES add 1172♦
```

A.5 OpenVMS VAX System Installation

VAX

The DECamds installation procedure consists of a series of questions and informational messages. See Example A–2 for a sample installation. The actual installation should take only a few minutes, but depending on your installation media, this time could be longer.

To abort the installation procedure at any time, press Ctrl/Y. When you press Ctrl/Y, the installation procedure deletes all files it has created up to that point and exits. You can then start the installation again.

To install DECamds:

1. **Invoke VMSINSTAL from a privileged account.**

```
@SYS$UPDATE:VMSINSTAL AMDS062 Device-name OPTIONS N
```

Device-name is the name of the device on which you plan to mount the media. For example, MTA0: is the device name for a tape drive.

OPTIONS N is an optional parameter that indicates you want to see the question on release notes. If you do not include the OPTIONS N parameter, VMSINSTAL does not ask you about the release notes. You should review the release notes before proceeding with the installation because they might contain additional information about the installation. If you are restarting the installation and have already reviewed the release notes, you do not need to specify OPTIONS N.

For more information about software installation options, see the *OpenVMS System Manager's Manual*.

2. **Confirm DECnet status.**

```
%VMSINSTAL-W-DECNET, Your DECnet network is up and running.
    .
    .
    .
* Do you want to continue anyway [NO]?
```

Type yes and continue the installation.

3. **Confirm system backup.**

```
* Are you satisfied with the backup of your system disk [YES]?
```

You should always back up your system disk before performing any installation. If you are satisfied with the backup of your system disk, press Return.

4. **Mount the media.**

```
Please mount the first volume of the set on MTA0:.
* Are you ready? YES
%MOUNT-I-MOUNTED, AMDS MOUNTED ON _MTA0:  (NODE 1)
The following products will be processed:
  DECamds V6.2

 Beginning installation of DECamds
V6.2 at 10:43

%VMSINSTAL-I-RESTORE, Restoring product saveset A...
```

You should now mount the first distribution volume on the device you specified when you invoked VMSINSTAL. The device name appears in the line preceding the question.

If you entered the wrong device name when you invoked VMSINSTAL and need to restart the installation, enter NO in response to the "Are you ready?" question. To abort the installation for other reasons, press Ctrl/Y.

If your installation kit contains more than one volume, VMSINSTAL prompts you to mount the additional volumes and then asks you to indicate that you are ready for the installation to proceed.

5. **Select a release notes option.**

   If you select Options N, the following is displayed:

```
Release notes included with this kit are copied to SYS$HELP.

Additional Release Notes Options:

 1.  Display release notes
 2.  Print release notes
 3.  Both 1 and 2
 4.  None of the above

* Select option [2]:
```

   If you choose to display the release notes, you can terminate the display at any time by pressing Ctrl/C. Release notes are copied to the system help directory (SYS$HELP:AMDS062.RELEASE_NOTES), even if you select option 4.

   After the installation, enter the following command to review the release notes through the Help facility:

```
$ HELP DECamds RELEASE_NOTES
```

   Note

   The name of the release notes file installed by VMSINSTAL consists of the current product name and version number. Do not delete release notes for previous versions of DECamds.

6. **Choose which components to install.**

```
You now have the choice to install either the DECamds Data Analyzer, or
the DECamds Data Provider, or both on this system.

* Would you like to install the DECamds Data Analyzer on this system [Y]?
* Would you like to install the DECamds Data Provider on this system [Y]?
```

7. **Respond to license registration queries.**

   You will be asked if you have loaded the appropriate license.

```
The right to use the DECamds Data Provider is granted with the
VAXCLUSTER license.  This installation will only ask for the
presence of the license.  If you answer yes to the following
question and do not have a license, then the IVP will fail.

* Do you have the VAXCLUSTER license loaded [Y]?
```

   If you have not registered and loaded your PAK, you must answer NO to this question. You must register and load your PAK, then restart the installation. For information on registering a license, see the *OpenVMS License Management Utility Manual*.

8. **Choose the Help Message installation option.**

```
The Help Message utility (MSGHLP) provides online explanations
and user actions for OpenVMS messages.

The MSGHLP database, DECAMDS.MSGHLP$DATA, consumes approximately
200 blocks and will be placed by default on your system disk in
SYS$COMMON:[SYSHLP].

Refer to the OpenVMS System Messages: Companion Guide for Help
Message Users and the OpenVMS System Manager manual for instruction
on how to add the DECamds Help Message database to your system.

* Would you like to install the MSGHLP database for DECamds  [Y]?
```

   If you answer YES (or use the default), the DECamds help message database, DECAMDS.MSGHLP$DATA, will be copied to SYS$COMMON:[SYSHLP]. Refer to the *OpenVMS System Messages: Companion Guide for Help Message Users* for complete information on setting up and using the Help Message utility.

9. **Choose the Installation Verification Procedure (IVP) option.**

```
* Do you want to run the IVP after the installation [YES]?
```

   It is recommended that you run the IVP because it ensures that the installation is successful. If the IVP finds the Data Provider on the system, it will ensure the driver is left in the state in which the IVP found it.

   You can run the IVP independently to verify that the software is available on your system. See Section A.9 for information about running the IVP separately.

10. **Choose the purge files option.**

```
* Do you want to purge files replaced by this installation [YES]?
```

    Purging is recommended; however, if you want to keep files from the previous version, enter NO and press Return.

11. **Read informational messages.**

    See the sample installation procedure in Example A–2.

12. **Observe the Installation Verification Procedure (IVP).**

    See the sample installation procedure in Example A–2.

Sample Installation on an OpenVMS VAX System

Example A–2 shows a sample installation on an OpenVMS VAX system and assumes that DECnet has been shut down, that no users are logged on to your system, and that OPTIONS N is specified to print the release notes. The Installation Verification Procedure (IVP) runs at the end of the installation.

This sample was run on a system that had no previous version of DECamds installed. Depending on which layered products you have on your system, you might see additional messages and questions when you perform your installation.

Example A–2 Sample OpenVMS VAX Installation

```
$ @VMSINSTAL AMDS062 SYS$UPDATE

 VAX/VMS Software Product Installation Procedure V5.5-2


It is 27-JAN-1995 at 15:13.

Enter a question mark (?) at any time for help.

* Are you satisfied with the backup of your system disk [YES]?

The following products will be processed:

  AMDS V6.2

 Beginning installation of AMDS V6.2 at 15:14

%VMSINSTAL-I-RESTORE, Restoring product save set A ...
%VMSINSTAL-I-RELMOVED, Product's release notes have been moved to SYS$HELP.

 Copyright (c) 1995 Digital Equipment Corporation. All rights reserved.

 Beginning the installation of DECamds V6.2.

%AMDS-I-VERINSTAL, Installing DECamds on an OpenVMS VAX V5 system

 You now may choose to install the DECamds Data Analyzer.  The
 Data Analyzer must be installed on a system in which at least
 DECwindows Motif V1.1 has been installed.

* Would you like to install the DECamds Data Analyzer on this system [Y]?

 The right to use the DECamds Data Provider is granted with the
 VAXCLUSTER license.  This installation will only ask for the
 presence of the license.  If you answer yes to the following
 question and do not have a license, then the IVP will fail.

* Do you have an appropriate license loaded [Y]?
* Do you want to run the IVP after the installation [YES]?
* Do you want to purge files replaced by this installation [YES]?

        No further questions will be asked during the installation.

 This installation will take approximately 5-10 minutes to complete.

%VMSINSTAL-I-RESTORE, Restoring product save set B ...

 The DECamds startup file (AMDS$STARTUP.COM) is being copied to the
 SYS$STARTUP directory.  Digital recommends you include this file in
 your site specific startup procedures as follows:

   @SYS$STARTUP:AMDS$STARTUP START

 The command procedure AMDS$LOGICALS.COM was already found in the
 SYS$COMMON:[AMDS] directory.  This installation will provide an
 AMDS$LOGICALS.TEMPLATE file instead of the AMDS$LOGICALS.COM file.
 Please check the new .TEMPLATE file and add any changes to your
 current AMDS$LOGICALS.COM file.

 The file SYS$COMMON:[AMDS]AMDS$CONSOLE_ACCESS.DAT already exists.  This
 installation will provide a AMDS$CONSOLE_ACCESS.TEMPLATE file instead
 of the AMDS$CONSOLE_ACCESS.DAT file to avoid superceding a version of
 the file you may currently be using.
```

(continued on next page)

Example A–2 (Cont.) Sample OpenVMS VAX Installation

```
  The installation will now provide the DECamds customization files.
  If this installation finds one of the customization files already
  residing in the installation directory, then the installation will
  provide a .TEMPLATE file instead of the .DAT file.  It will be left
  up to the installer to check the AMDS$SYSTEM directory for any
  differences between the .DAT and .TEMPLATE files.

  The installation procedure will provide a .TEMPLATE file for the
  following customization files found in the AMDS$SYSTEM directory:

  AMDS$APPLIC_CUSTOMIZE.DAT
  AMDS$SEVERITY_DEFS.DAT
  AMDS$OCCURRENCE_DEFS.DAT
  AMDS$THRESHOLD_DEFS.DAT
  AMDS$INTERVAL_DEFS.DAT
  AMDS$FILTER_DEFS.DAT
  AMDS$SORT_DEFS.DAT
  AMDS$GEOMETRY_DEFS.DAT

  The file SYS$COMMON:[AMDS]AMDS$DRIVER_ACCESS.DAT already exists.  This
  installation will provide a AMDS$DRIVER_ACCESS.TEMPLATE file instead
  of the AMDS$DRIVER_ACCESS.DAT file to avoid superceding a version of
  the file you may currently be using.


  DECamds installation V6.2 is complete.

%VMSINSTAL-I-MOVEFILES, Files will now be moved to their target directories...
%AMDS-I-NODRIVER, data collection not started, only define AMDS$SYSTEM

  DECamds V6.2  Installation Verification Procedure (IVP) Beginning.

  Copyright © Digital Equipment Corporation 1995.
  All Rights Reserved.

This IVP will check following pieces of DECamds:

     DATA PROVIDER - RMDRIVER

     DATA ANALYZER - AMDS$CONSOLE and AMDS$COMM

Before running this IVP, any process currently running the DECamds
Data Analyzer on this machine must be stopped.  The IVP will now
check for the existence of the AMDS$COMM process which indicates
that someone is currently running the Data Analyzer.

AMDSIVP-I-RMRELOAD, reloading with new data provider for IVP to use

Copyright (c) 1995 Digital Equipment Corporation. All rights reserved.
%AMDS-I-RMSHUT, stopping RMDRIVER processing for this node
%AMDS-S-RMSUCCESS, RMDRIVER shutdown successful


Copyright (c) 1995 Digital Equipment Corporation. All rights reserved.
%AMDS-I-RMSTART, starting RMDRIVER processing for this node
%AMDS-I-LOADSECDB, loading security database
%AMDS-S-RMSUCCESS, RMDRIVER startup successful


You currently do not have a DECwindows display to direct the Data
Analyzer output towards.  You are now given the option to continue
and allow the IVP to create a display for you or exit and either
create your own display or run the IVP from a workstation.
```

(continued on next page)

Example A–2 (Cont.) Sample OpenVMS VAX Installation

```
Would you like to continue? [Y/N]Y

You have chosen to let the IVP create a DECwindows display.
You will be prompted for a node to direct the display towards.
Specify a valid node on which you are allowed to create a display
from your current node.  This procedure does not check the
validity of the remote node you specify.

Nodename to direct the display towards: AMDS
%DCL-S-SPAWNED, process AMDS$IVP_CONS spawned

Waiting 5 minutes to allow the DECamds Data Analyzer to run.
You should see the Event Log and System Overview windows appear
on your DECwindows display device.

AMDSIVP-S-ALLPASS, DECamds has passed the IVP

 DECamds V6.2  Installation Verification Procedure Complete

 Installation of AMDS V6.2 completed at 15:23

 VMSINSTAL procedure done at 15:23

$ LOG
  SYSTEM       logged out at 27-JAN-1995 15:23:53.60♦
```

Recovering from Errors

Errors can occur during the installation if any of the following conditions exist:

- The operating system version is incorrect.
- A prerequisite software version is incorrect.
- Privileges necessary for successful IVP are insufficient.
- Quotas necessary for successful IVP are insufficient.
- System parameter values for successful IVP are insufficient.
- The OpenVMS help library is currently in use.
- The product license has not been registered and loaded.
- The node on which the DECamds Data Analyzer was being displayed does not allow connections from your process and node.

For descriptions of the error messages generated by these conditions, see the OpenVMS documentation on system messages, recovery procedures, and OpenVMS software installation. If you are notified that any of these conditions exist, you should take the appropriate action as described in the message. For information on installation requirements, see Section A.3.

If the IVP fails, check the AMDS$SYSTEM:AMDS$CONS_LOG.LOG for any errors that might have occurred while the IVP was executing the console verification portion.

A.6 Postinstallation Tasks on Data Provider Nodes

Perform the following tasks after installing DECamds on Data Provider nodes:

1. If you have not read the release notes, please read them.
2. Modify user accounts.

   Users who maintain the security or group name files or load new versions of the driver need privileges associated with the driver startup procedure.
3. Add AMDS$STARTUP.COM to the node's startup and shutdown procedures to provide for automatic startup and shutdown of the Data Provider driver when a node is booted or shut down.

   - On an OpenVMS Version 5.5 system, add the following command line to SYS$MANAGER:SYSTARTUP_V5.COM

     ```
     $ @SYS$STARTUP:AMDS$STARTUP.COM START
     ```

   - On any OpenVMS system, add the following command line to SYS$MANAGER:SYSTARTUP_VMS.COM

     ```
     $ @SYS$STARTUP:AMDS$STARTUP.COM START
     ```

   Also, edit SYSHUTDWN.COM to add the following line:

   ```
   $ @SYS$SHUTDOWN:AMDS$STARTUP.COM STOP
   ```
4. Modify default security files.

   To implement fixes, which require write access, the security files must be modified. The Data Provider security file contains a list of three-part codes representing Data Analyzer nodes that have read or write access to that node. Refer to Section 1.3 for complete instructions about designing security files.
5. Assign a node to a group.

   See Section 1.3.2.1.
6. Start DECamds (the Data Provider).

   Even though the IVP starts and stops the driver, you must start the Data Provider drivers by entering the following command on each node:

   ```
   $ @SYS$STARTUP:AMDS$STARTUP.COM START
   ```

Note

Starting, stopping, or reloading DECamds (the AMDS$STARTUP.COM procedure) requires at least TMPMBX, NETMBX, SYSNAM, LOG_IO, and CMKRNL privileges. Use the OpenVMS Authorize utility (AUTHORIZE) to determine whether users have the required privileges and then make adjustments as needed.

A.6.1 Starting, Stopping, and Reloading DECamds

To start and stop the Data Provider driver, enter the following command: (Use this command if a node will be used to both provide and collect system data.)

```
$ @SYS$STARTUP:AMDS$STARTUP.COM [parameter]
```

where the optional *parameter* is one of the following:

NODRIVER	Defines the default input and output logicals on the Data Analyzer node driver. Use this parameter on the Data Analyzer node where the Data Provider driver is not running. It is the default.
START	Starts the Data Provider driver.
STOP	Stops the Data Provider driver.
RELOAD	Loads a new Data Provider driver. Use this parameter when installing a new version of DECamds.

Note

On systems running OpenVMS VAX Version 6.0, if you use the OpenVMS Snapshot Facility, stop the DECamds Data Analyzer and Data Provider node drivers before taking a system snapshot.

A.7 Postinstallation Tasks on the Data Analyzer Node

Perform the following tasks after installing the DECamds Data Analyzer:

1. If you were previously running an earlier version of DECamds, check the differences between the .DAT or .COM files on your system and the associated .TEMPLATE files provided with the new kit. Change your existing files as necessary.

   Note

   The new .TEMPLATE files may contain important changes. However, to avoid altering your customizations, the upgrade procedure does not modify your existing customized versions of these files. Check the new .TEMPLATE versions of these files provided with the kit, and make the appropriate change to your files.

2. Modify default DECamds security files on each Data Analyzer node.

   To implement fixes, which require write access, the security files must be modified. Refer to Section 1.3 for complete instructions about designing security files.

3. Define the system directory logical name AMDS$SYSTEM.

   To define the logical name AMDS$SYSTEM on systems running the Data Analyzer but not the Data Provider, enter the following command:

   ```
   $ @SYS$STARTUP:AMDS$STARTUP.COM NODRIVER
   ```

   This command requires SYSNAM privilege. The NODRIVER parameter specifies that the procedure is to define the input and output logical names in AMDS$LOGICALS.COM.

4. Modify user accounts as needed.

To use DECamds, user accounts require certain privileges and quotas:

- Using the Data Analyzer node for data collection (read access) requires TMPMBX, NETMBX, and OPER privileges.
- Performing fixes (write access) requires the CMKRNL privilege in addition to TMPMBX, NETMBX, and OPER.
- Using the AMDS$STARTUP.COM to start, stop, or reload the Data Provider requires at least TMPMBX, NETMBX, SYSNAM, LOG_IO, and CMKRNL privileges.

5. Start the application.

   For example, the following command starts DECamds with all input files read from AMDS$SYSTEM and all output files written to the current default directory. Only data from group A nodes and group B nodes are displayed.

```
$ AVAIL /CONFIGURE=AMDS$SYSTEM /LOG_DIRECTORY=SYS$LOGIN-
_$/GROUP=(GROUP_A, GROUP_B)
```

   See Chapter 2 for startup options.

A.8 Determining and Reporting Problems

If you encounter a problem while using DECamds, report the problem to Digital. Depending on the nature of the problem and the type of support you have, take one of these actions:

- If your software contract or warranty agreement entitles you to telephone support, call your Digital support center.
- If the problem has to do with the DECamds documentation, fill out and submit a Reader's Comments form (from the back of the manual). Use the form from the manual in which you found the error. Include the section and page number.

Review the Software Product Description (SPD) and Warranty Addendum for an explanation of the warranty. If you encounter a problem during the warranty period, report the problem as indicated above or follow alternate instructions provided by Digital for reporting SPD nonconformance problems.

A.9 Running the Installation Verification Procedure Separately

Usually the Installation Verification Procedure (IVP) runs during installation. Should system problems occur after you have installed DECamds, check the integrity of installed files by executing the following command procedure:

```
$ @SYS$TEST:AMDS$IVP.COM
```

The IVP leaves the Data Provider in the same state in which it was found. For example, if the Data Provider is running, the IVP stops and starts it.

B

Files and Logical Names

The DECamds installation procedure installs files and defines logical names to customize the environment.

The installation procedure defines all logical names in executive mode in the system table (with the /SYSTEM /EXECUTIVE qualifiers) However, you can define logical names in /JOB or /GROUP tables, preceding the system definitions.

Table B–1 and Table B–2 explain the files installed and logical names defined with the Data Analyzer.

Table B–3 and Table B–4 explain the files installed and logicals defined on each node running the Data Provider.

Logical names are added to the logical name table when the AMDS$LOGICALS.COM procedure is invoked by AMDS$STARTUP.COM.

Note

Logical names can be a search list of other logicals.

The logical names in Table B–4 and Table B–2 must be defined in the job, group, or system table. If you change the name, define the new logical in the job, group, or system table.

B.1 Files and Logical Names for the Data Analyzer Node

Table B–1 and Table B–2 contain the names of all files created on a Data Analyzer node when DECamds is installed.

Table B–1 Files on the DECamds Data Analyzer

Directory-Logical:File-Name	Function
AMDS$HELP:AMDS$HELP.HLB	Help library
AMDS$SYSTEM:AMDS$*.DAT	Customization files
AMDS$SYSTEM:AMDS062.RELEASE_NOTES	Product Release Notes
AMDS$SYSTEM:AMDS$COMM.EXE	Communication image
AMDS$SYSTEM:AMDS$CONSOLE.EXE	Data Analyzer image
AMDS$SYSTEM:AMDS$CONSOLE.UID	User interface description file
AMDS$SYSTEM:AMDS$CONSOLE_ ACCESS.DAT[1]	Data Analyzer security file

[1]Can be provided as a TEMPLATE file, depending on whether the file was found during installation.

(continued on next page)

Table B–1 (Cont.) Files on the DECamds Data Analyzer

Directory-Logical:File-Name	Function
AMDS$SYSTEM:AMDS$LOGICALS.COM[1]	Logical name definition file
AMDS$SYSTEM:AMDS$VMS*-*.LIB	DECamds version-specific libraries
AMDS$TEST:AMDS$IVP.COM	Installation verification procedure
SYS$STARTUP:AMDS$STARTUP.COM	DECamds startup file

[1]Can be provided as a TEMPLATE file, depending on whether the file was found during installation.

Table B–2 Logical Names Defined for the Data Analyzer

Logical Name	Definition	Default
AMDS$COMM_BUFFER_ SIZE	This value is the size (in bytes) of the communications buffer between the AMDS$CONSOLE process and the AMDS$COMM process.	300000 bytes
AMDS$COMM_READS	This value is the number of read aheads posted by the DECamds communications process (AMDS$COMM) to handle the delivery of remote response packets from the Data Provider to the Data Analyzer node.	50 read aheads
AMDS$COMM_PKT_RETRY	Specifies the number of retries before quitting and issuing a "delivery path lost message."	4
AMDS$COMM_PKT_ TMOUT	Specifies the timeout period (in seconds) for packet retry for the Data Analyzer.	10
AMDS$CONFIG	The device and directory location for the following DECamds input files: AMDS$APPLIC_CUSTOMIZE.DAT AMDS$COMM.EXE AMDS$CONSOLE.UID AMDS$CONSOLE_ACCESS.DAT AMDS$VMS*-*.LIB All customization files AMDS$*_ DEFS.DAT	AMDS$SYSTEM
AMDS$DPI	This value specifies the DPI value of your display device.	75 or 100
AMDS$LOG	The device and directory location for the following DECamds output files: AMDS$EVENT_LOG.LOG AMDS$LOCK_LOG.LOG	AMDS$SYSTEM

B.2 Files and Logical Names for the Data Provider Nodes

Table B–3 and Table B–4 contain the names of all files created on a node when a Data Provider is installed.

Table B–3 Files on Nodes Running the Data Provider

Directory-Logical:File-Name	Function
AMDS$SYSTEM:AMDS$DRIVER_ ACCESS.DAT[1]	Data Provider security file
AMDS$SYSTEM:AMDS$LOGICALS.COM[1]	Logical name definition file
AMDS$SYSTEM:RMCP.EXE	Management interface to the Data Provider
SYS$HELP:AMDS062.RELEASE_NOTES	Product Release Notes
SYS$HELP:AMDS$HELP.HLB	Help library
SYS$LOADABLE_IMAGES:RMDRIVER.EXE, .STB[2]	On VAX systems, Data Provider
SYS$LOADABLE_ IMAGES:SYS$RMDRIVER.EXE, .STB[3]	On Alpha systems, OpenVMS Alpha driver.
SYS$STARTUP:AMDS$STARTUP.COM	DECamds startup file
SYS$TEST:AMDS$IVP.COM	Installation verification procedure

[1]Can be provided as a TEMPLATE file, depending on whether the file was found during installation.

[2]VAX specific

[3]Alpha specific

Table B–4 Logical Names Defined on Nodes Running the Data Provider

Logical Name	Definition	Default
AMDS$CONFIG	The device and directory location for the DECamds input file AMDS$DRIVER_ ACCESS.DAT.	AMDS$SYSTEM
AMDS$DEVICE	This logical is translated as the first LAN device to which the Data Provider or Data Analyzer node attempts to connect. The attempts are made in this order: AMDS$DEVICE, FXA0, XEA0, XQA0, EFA0, ETA0, ESA0, EXA0, EZA0, FCA0, ECA0. If your LAN line is not in this list, use AMDS$DEVICE. If the Data Analyzer node and Data Provider run on the same node, *RMA0 is used.	Undefined
AMDS$GROUP_NAME	The Group to which the node is assigned. Choose an alphanumeric string of up to 15 characters. The Group Name is defined on the node running the Data Provider and is used by the Data Analyzer node to display nodes in the System Overview window.	DECamds

(continued on next page)

Table B–4 (Cont.) Logical Names Defined on Nodes Running the Data Provider

Logical Name	Definition	Default
AMDS$NUM_DL_READS	The number of data link reads to be posted by the Data Provider as read ahead buffers. Generally between 4 and 8 should be sufficient to allow the Data Provider to process without having to wait for a data link buffer to be cleared.	5 data link reads[1]
AMDS$RM_DEFAULT_ INTERVAL	The number of seconds between multicast hello messages from the Data Provider to the Data Analyzer node when the Data Provider is inactive or is only minimally active. The minimum value is 15. The maximum value is 300.	30
AMDS$RM_OPCOM_READ	This logical defined as TRUE allows OPCOM messages for READ failures from the Data Provider. Defined as FALSE, the message facility is disabled.	TRUE
AMDS$RM_OPCOM_ WRITE	This logical defined as TRUE allows OPCOM messages for write (Fix) successes and failures from the Data Provider. Defined as FALSE, the message facility is disabled.	TRUE
AMDS$RM_SECONDARY_ INTERVAL	The number of seconds between multicast hello messages from the Data Provider to the Data Analyzer node when the Data Provider is active. The minimum value is 15. The maximum value is 1800.	90

[1]Each read request requires 1500 bytes of BYTCNT quota used for the starting process.

B.3 Log Files

The DECamds Data Analyzer records two log files:

- An events log file named AMDS$EVENT_LOG.LOG. This ASCII text file records all event messages displayed in the Event Log window.
- A lock contention log file named AMDS$LOCK_LOG.LOG. This ASCII text file records all lock contention information displayed in the Lock Contention window.

Both log files are created when the DECamds application is started. Either file can be edited while the application is running.

B.4 Event Log File

The event log file keeps a record of the events detected by DECamds. You can review it without a DECwindows terminal. Every 30 minutes, DECamds writes a message to the file, noting the date and time.

Example B–1 is an example of AMDS$LOG:AMDS$EVENT_LOG.LOG.

Example B–1 Sample Event Log File

```
Time      Sev   Event
Opening DECamds Event Log on date/time: 11:16:07.98
11:16:07.98 0  CFGDON, PROD12 configuration done
11:16:08.44 0  CFGDON, PROD09 configuration done
11:16:09.65 0  CFGDON, AXPND1 configuration done
11:16:11.47 0  CFGDON, PROD01 configuration done
11:16:11.89 0  CFGDON, VAXND1 configuration done
11:16:12.14 0  CFGDON, PROD15 configuration done
11:16:14.02 0  CFGDON, PROD14 configuration done
11:16:14.57 60 HIDIOR, PROD12 direct I/O rate is high
11:16:14.57 70 HITTLP, PROD12 total page fault rate is high
11:16:14.57 80 LOMEMY, PROD12 free memory is low
11:16:14.58 70 HITTLP, PROD09 total page fault rate is high
11:16:14.58 80 LOMEMY, PROD09 free memory is low
11:16:15.32 70 HITTLP, AXPND1 total page fault rate is high
11:16:25.33 60 HIBIOR, PROD09 buffered I/O rate is high
11:16:35.46 60 HIBIOR, AXPND1 buffered I/O rate is high
11:16:40.62 95 LOSWSP, AXPND1 DISK$ALPHAVMS015:[SYS0.SYSEXE]SWAPFILE.SYS swap file space is low
11:16:49.84 70 HITTLP, PROD09 total page fault rate is high
11:16:55.14 60 HIBIOR, PROD12 buffered I/O rate is high
11:17:14.58 0  CFGDON, PROD05 configuration done
11:17:14.94 70 HITTLP, PROD09 total page fault rate is high
11:17:16.93 0  CFGDON, PROD04 configuration done
11:17:18.10 0  CFGDON, PROD17 configuration done
11:17:18.15 0  CFGDON, PROD10 configuration done
11:17:19.50 60 HIBIOR, PROD10 buffered I/O rate is high
11:17:19.50 60 HIDIOR, PROD10 direct I/O rate is high
11:17:19.50 70 HITTLP, PROD10 total page fault rate is high
11:17:19.50 80 LOMEMY, PROD10 free memory is low
11:17:20.33 60 HIBIOR, PROD05 buffered I/O rate is high
11:17:21.49 0  CFGDON, PROD20 configuration done
11:17:21.52 0  CFGDON, PROD13 configuration done
11:17:24.96 0  CFGDON, PROD06 configuration done
11:17:35.35 0  CFGDON, PROD07 configuration done
11:17:39.84 60 HINTER, PROD07 interrupt mode time is high
11:17:40.21 70 HITTLP, PROD09 total page fault rate is high
11:18:04.69 60 HIBIOR, PROD10 buffered I/O rate is high
11:18:05.36 60 HIDIOR, PROD07 direct I/O rate is high
11:18:10.49 60 HIBIOR, PROD09 buffered I/O rate is high
11:18:10.49 60 HIDIOR, PROD09 direct I/O rate is high
11:18:14.70 60 HIBIOR, PROD12 buffered I/O rate is high
11:18:15.68 60 HIBIOR, AXPND1 buffered I/O rate is high
11:18:26.05 60 HIBIOR, PROD05 buffered I/O rate is high
11:18:40.57 75 HIHRDP, PROD10 hard page fault rate is high
11:18:45.80 60 HIDIOR, PROD09 direct I/O rate is high
11:18:55.91 60 HINTER, PROD07 interrupt mode time is high
11:19:09.67 60 HIBIOR, PROD09 buffered I/O rate is high
11:19:09.67 60 HIDIOR, PROD09 direct I/O rate is high
11:19:09.67 75 HIHRDP, PROD09 hard page fault rate is high
11:19:15.48 60 HIBIOR, PROD05 buffered I/O rate is high
```

B.5 Lock Contention Log File

Example B–2 is an example of a Lock Contention Log File.

Example B–2 Sample Lock Contention Log File

```
*********************************************

 Time:  9-JUL-1993 14:23:46.68
  Master Node: AXPND1
  Resource Name: QMAN$JBC_ALIVE_01
  Parent Resource Name: QMAN$MSR_$10$DKA300.....ñ.....
  RSB Address: 805B1400, GGMODE: EX, CGMODE: EX
  Hex Representation
    514D414E   244A4243   (Bytes 0 - 7)
    5F414C49   56455F30   (Bytes 8 - 15)
    31000000   00000000   (Bytes 16 - 23)
    00000000   000000C0   (Bytes 24 - 31)
  Status: VALID
*********************************************

 Time:  9-JUL-1993 14:28:42.44

  Resource Name: QMAN$JBC_ALIVE_01
  Parent Resource Name: QMAN$MSR_$10$DKA300.....ñ.....
  Blocking Lock Data
   Node: AXPND1, PID: 2020008C, Name: JOB_CONTROL
   LKID: 0200015E, GR Mode: EX
   Flags: NOQUEUE
   Local Copy
  Blocked Lock on WAITING queue
   Node: AXPND1, PID: 2020008D, Name: QUEUE_MANAGER
   LKID: 2000013B, RQ Mode: CR
   Flags: NODLCKW
   Local Copy
*********************************************
```

B.6 OPCOM Log

The following examples show some of the OPCOM messages that appear in the operator log file from the Data Provider:

```
%%%%%%%%%%  OPCOM   2-JAN-1992 08:16:21.92  %%%%%%%%%%
Message from user RMDRIVER
RMA0: - No privilege to access from node 2.2
```

This means that the node does not have the privilege to perform a read operation.

```
%%%%%%%%%%  OPCOM   2-JAN-1992 10:10:45.08  %%%%%%%%%%
Message from user RMDRIVER
RMA0: - No privilege to write from node 2.2
```

This means that the Data Provider does not have the privilege to perform a write operation.

```
%%%%%%%%%%  OPCOM   2-JAN-1992 12:35:05.28  %%%%%%%%%%
Message from user RMDRIVER
RMA0: - Process 2390003c modified from node 2.2
```

This means that the Data Provider has successfully performed a WRITE operation on the node.

Glossary

Automatic Data Collection

Data collection that begins automatically when the Data Analyzer runs and recognizes a Data Provider. By default, this feature is enabled.

The default data windows for which automatic collection is enabled are:

Node Summary
Page/Swap File Summary
Lock Contention Summary

Automatic Event Collection

Allows the event detection code to automatically start collecting more data to try and determine the root cause of an event. Automatic investigation only applies to events that are detected after you enable the option.

This feature does not apply to any Lock Contention events. To enable automatic lock contention detection, use the DECamds Customizations dialog box, as explained in Section 5.1.

Collection Interval

The frequency at which the Data Analyzer will send requests to a Data Provider to collect data.

Data Analyzer

The portion of DECamds that collects and displays system data from Data Provider nodes. You can also perform fixes with the Data Analyzer.

Data Provider

The portion of DECamds that is installed for purposes of providing system data when requested by authorized Data Analyzers. A Data Provider node uses the OpenVMS LAN drivers to receive and send data across the network.

Data Window

A Data Analyzer window that displays additional data. Several different data windows are available as follows (see also Chapter 3):

Node Summary
CPU Modes Summary
CPU Summary
Memory Summary
Process I/O Summary
Single Process Summary
Disk Status Summary
Disk Volume Summary
Page/Swap File Summary

Lock Contention Summary
Single Lock Summary

Event

A description of a potential resource availability problem, based on rules defined by the Data Analyzer and customized thresholds. Events trigger display changes in data windows such as color and item highlighting.

Event Log Window

One of two primary Data Analyzer windows that displays events as they occur. For each event, you can display more detailed information to investigate the underlying problem by double clicking on the event. You can also perform fixes for some events from this window.

Fix

A corrective action made to a Data Provider node but initiated from the Data Analyzer node.

Group

A set of remote Data Provider nodes with similar attributes, for example all the members of a VMScluster can be in the same group. The group that a node belongs to is determined by the translation of the AMDS$GROUP_NAME logical on each of the Data Analyzers.

Page

A unit used by the operating system to section memory. On VAX systems, a page is 512 bytes. On Alpha systems, a page may be 8 kilobytes (8192 bytes), 16 KB, 32 KB, or 64 KB.

Pagelet

A unit used by the OpenVMS Alpha operating system to break down the page into smaller addressable units. One pagelet is the same as a VAX page, 512 bytes.

System Overview Window

One of two primary Data Analyzer windows that graphically displays all groups and the nodes that belong to each group. The System Overview window provides summary data about CPU, Memory, and Process I/O usage for the nodes being monitored.

Security Triplet

A three-part access code located in the AMDS$DRIVER_ACCESS.DAT and AMDS$CONSOLE_ACCESS.DAT files that enables communications to be established between the Data Analyzer and Data Provider.

Index

A

B

C

D

E

F

G

H

N

O

P

Q

R

S

T

U

V

W

AA-Q3JSB-TE

Printed in Europe

Printed on Environmentally Friendly Paper